



UNIVERSIDAD CARLOS III DE MADRID

## **TESIS DOCTORAL**

# **Diseño, Análisis y Evaluación de Conjuntos de Clasificadores basados en Redes de Neuronas**

**Autor:** M. Paz Sesmero Lorente

**Directores:** Dra. Araceli Sanchis de Miguel  
Dr. Germán Gutiérrez Sánchez

**DEPARTAMENTO DE INFORMÁTICA**

Leganés, Septiembre de 2012



# TESIS DOCTORAL

## DISEÑO, ANÁLISIS Y EVALUACIÓN DE CONJUNTOS DE CLASIFICADORES BASADOS EN REDES DE NEURONAS

Autor: M. Paz Sesmero Lorente

Director/es: Araceli Sanchis de Miguel  
Germán Gutiérrez Sánchez

Tribunal Calificador:

Firma

Presidente: .....

Vocal: .....

Secretario: .....

Calificación: .....

Leganés, ..... de ..... de 2012



*La vida es una serie de colisiones con el futuro;  
no es una suma de lo que hemos sido, sino de lo que anhelamos ser.*

José Ortega y Gasset



# Agradecimientos

Cuando llega el momento de escribir estas líneas se abre una ventana en el tiempo que te hace recordar a todas aquellas personas que, de una u otra manera, han contribuido a la realización de este Trabajo. Algunas de ellas estuvieron en el principio, otras surgieron a medio camino y otras han estado desde el principio hasta el fin. A todos vosotros, gracias.

A mis padres, porque, aunque sé que os costó, me permitisteis venir a Madrid, por apoyarme cuando os dije que quería hacer el Doctorado, por las veces que me habéis preguntado cómo me iba y por aquellas veces que habéis optado por no preguntarlo, por respetar mis ausencias, pero, sobre todo, por confiar en mí.

A Juan, por compartir conmigo los mejores y los peores momentos, por apoyar mis decisiones, por conseguir que me levantara cuando ya lo daba todo por perdido, por intentar entenderme cuando ni yo misma me entendía y sobre todo, por querer ser mi compañero en este viaje que es la vida.

A mis directores de Tesis, Araceli y Germán. Por asesorarme, por guiarme en este largo camino, por soportar el pesimismo con el que algunas veces he juzgado mi trabajo y por hacerme ver que, al margen de lo que yo pensara, gran parte del trabajo podía formar parte de la Tesis.

A Agapito, por todos los consejos que me has dado, por tu pragmatismo y capacidad de síntesis, por atenderme cada vez que lo he requerido y, sobre todo, por esa frase que marcó el principio del fin: *esto también es investigación*.

A los chicos de CAOS, y en especial a mis *jotas*. A Jose, por transmitirme parte de tu optimismo y alegría, por ser mi cómplice en muchas ocasiones, por cuidarme y protegerme en esos momentos que tú y yo sabemos. Por ser mi amigo. A JMAW, porque sin ti, esta Tesis nunca hubiera sido posible. Por infundirme parte de tus conocimientos, por ayudarme en los momentos fáciles y, también, en los difíciles, por creer en mí cuando yo no creía y, en definitiva, por todo lo que me has asesorado y todo el tiempo que desinteresadamente me has dedicado. A Jorge, mi *compi* de Laboratorio, y a Javi, mi *compi* de docencia, por todo el trabajo que me habéis quitado y por todas esas conversaciones que tanto me han aportado. Gracias también a ti, Raúl, porque cuando te he necesitado has respondido y cuando te he *reprobado* lo has entendido.

A las *ladies*, Paula y Beatriz, por vuestros consejos y opiniones, por respetar mis silencios, por romperlos cuando ha sido necesario, por mostrarme diferentes perspectivas de la vida, por ser como sois. Gracias también a Cris, mi amiga gallega, porque siempre confiaste en que llegaría este día.

A las *jotas* de mis otras vidas, Joaquín Estremera y Jesús Reviejo, por escucharme, por aconsejarme, por creer que sí podía. A Juan Antonio Maestro, por darme la oportunidad que me diste y conseguir que recuperara la autoestima.

A Lola, por introducirme en el mundo del *Machine Learning*, por demostrarme que un Físico también puede dedicarse a estos temas y por todos esos granitos de arena que has aportado a mi formación.

Por último, pero no por ello menos importante, a mi familia *Alicantina*, por tratarme como a uno un más, por los ratos tan agradables que me hacéis pasar y por haber comprendido que en esta etapa final necesitaba un respiro. A Fernan, porque, aunque fuera en la sombra, sé que siempre has estado pendiente de cómo me iba. A mis *físicos* y a mis *niñas de la resi*, por estar ahí desde el principio, por perdonar mis ausencias y por intentar entenderlas. Gracias por formar parte de mi vida y por dejar que forme parte de la vuestra.



## Resumen

Una de las áreas de investigación que, dentro del marco del *Aprendizaje Automático*, más atención ha recibido durante las últimas décadas ha sido el diseño de conjuntos de clasificadores. Bajo este denominador se engloban un gran número de algoritmos cuyo objetivo es la construcción de un clasificador robusto haciendo uso de clasificadores más simples denominados clasificadores base. Aunque el uso de los conjuntos de clasificadores se puede argumentar desde diversas perspectivas, la justificación más evidente se encuentra en el comportamiento humano. Antes de tomar una decisión importante es habitual pedir opinión a varios expertos para así tener mayor certeza de que la opción elegida es la más adecuada.

Diversos estudios han demostrado que el éxito de cualquier conjunto de clasificadores viene determinado por la *precisión* y la *diversidad* de los clasificadores que lo integran. En otras palabras, para que un conjunto de clasificadores mejore la precisión de cualquiera de sus miembros se requiere que éstos sean precisos y diversos. Sin embargo, encontrar clasificadores base que, de forma simultánea, satisfagan ambos requisitos no es una tarea fácil. Por ello, en este trabajo se presentan dos nuevas arquitecturas de conjuntos de clasificadores en una de las cuales, sin obviar la diversidad, se fomenta la precisión de los clasificadores base, mientras que en la otra se fomenta la diversidad frente a la precisión. Las diferencias y la complementariedad existente entre ambas arquitecturas permitirá analizar la influencia que, en el comportamiento global del conjunto, tiene la primacía de una de estas particularidades frente a la otra.

Aunque, en el mundo real, la mayor parte de los problemas de clasificación engloban a más de dos categorías, muchos de los conjuntos de clasificadores propuestos en la Bibliografía fueron originalmente concebidos para resolver problemas dicotómicos. En ocasiones, el algoritmo que rige el comportamiento de estos modelos puede extrapolarse a problemas *multiclase*. Sin embargo, en otros muchos casos, el problema *multiclase* sólo se puede resolver descomponiendo el problema original en subproblemas binarios. Además, la mayor parte de los modelos propuestos, han sido evaluados sobre dominios artificiales en los que el número de atributos con los que se describen los ejemplos es relativamente pequeño. A pesar de esta tendencia, existen un gran número de dominios reales en los que los ejemplos están descritos por cientos o incluso miles de características. La necesidad de disponer de nuevos métodos de clasificación capaces de resolver *problemas reales* marca uno de los objetivos de esta Tesis Doctoral. Así, las arquitecturas que se proponen en este trabajo han sido concebidas explícitamente para la resolución de problemas en los que el número de categorías es finito y superior a dos y en los que los ejemplos están descritos por un elevado número de atributos. Partiendo de estas dos singularidades, se pretende acotar, en la medida de lo posible, la complejidad y el coste computacional inherentes a la resolución de este tipo de problemas.

La viabilidad de las arquitecturas propuestas se ha determinado experimentalmente. Así, el estudio realizado contempla un exhaustivo análisis en el que, sobre distintos dominios, se analiza el comportamiento de las arquitecturas propuestas y se compara con el logrado por algunos de los modelos de clasificación más referenciados en la Bibliografía.

# Abstract

The design of *Ensemble of Classifiers* has been one of the most active research areas in the field of *Machine Learning* for the last decades. In this area, many different algorithms have been proposed in order to create a more robust classifier that consists of simpler classifiers named base classifiers. Although the use of ensemble of classifiers can be justified by many different reasons, the most obvious justification is related with human decision making process. Before making a decision, it is common to ask several experts to be sure that the chosen option is the optimal.

Many studies have demonstrated that the success of any ensemble of classifiers is related to the *accuracy* and *diversity* of the different base classifiers of the ensemble. In other words, an ensemble of classifiers could improve the accuracy of any of its individual members if they are accurate and diverse. However, obtaining base classifiers which satisfy both requirements simultaneously is not an easy task. For this reason, this work presents two new ensembles of classifiers: One of these ensembles prioritizes the accuracy of the base classifiers (taking also into account the diversity) and the other promotes diversity over accuracy. These ensembles are different but complement each other, so it will be possible to analyze the influence of these requirements over the global performance of the ensemble.

The number of applications that require multiclass categorization is huge in the real world. However, many of the studies related to supervised learning are focused on the resolution of binary problems. Some machine learning algorithms can then be naturally extended to handle the multiclass case. For other algorithms, a direct extension to the multiclass case may be problematic. Typically, in such cases, the multiclass problem is reduced to multiple binary classification problems that can be solved separately.

In addition, most of these models have been evaluated in artificial domains in which the number of features used to describe the examples is relatively small. Despite this, there are many real domains in which the examples are described by hundreds or even thousands of features. For this reason, one of the goals of this thesis is the creation of new classification methods for real world. Thus, the ensembles proposed in this work have been designed to be applicable to real domains in which each example is labeled with one of several categories and is described by a large number of features. Taking these characteristics into account, the computational complexity and cost of the proposed methods need to be reduced as much as possible.

The viability of the proposed ensembles has been proved empirically. Thus, this thesis makes a comprehensive analysis in which, taking into account different domains, the performance of the proposed ensembles is analyzed and compared with other well-known classification methods.



# Índice General

<b>1. Introducción</b>	1
<b>2. Estado del Arte</b>	5
2.1. Conjunto de Clasificadores	5
2.2. Fase de generación de los clasificadores base	6
2.2.1. Generación de clasificadores diversos	7
2.2.2. Medidas de diversidad	9
2.2.3. Optimización del grupo de clasificadores base	12
2.3. Fase de integración de los clasificadores base	15
2.4. Estudios comparativos	19
<b>3. Objetivos de la Tesis Doctoral</b>	23
<b>4. Descripción de los Modelos Propuestos: BCE y CCE</b>	27
4.1. Introducción	28
4.2. Descripción de BCE	30
4.2.1. Construcción de los módulos base	30
4.2.2. Arquitectura de los módulos base: Integración Binario-Complementario.	35
4.3. Descripción de CCE	37
4.3.1. Características de la arquitectura: Módulos base	38
4.3.2. Optimización de los módulos base	40
4.4. Integración de los clasificadores base	43
<b>5. Validación Experimental</b>	45
5.1. SYNTHETIC: Seis clases y conjunto de entrenamiento reducido	46
5.1.1. Evaluación de BCE	47
5.1.2. Evaluación de CCE	57
5.2. SEMEION: Diez clases y pocos ejemplos de entrenamiento	60
5.2.1. Evaluación de BCE	61
5.2.2. Evaluación de CCE	64

5.3.	SEMEION MODIFICADO: Desbalanceo entre clases.....	68
5.4.	MNIST: Diez clases y gran número de ejemplos de entrenamiento .....	71
5.4.1.	Procedimiento experimental.....	72
5.4.2.	Características de los Modelos Evaluados.....	74
5.4.3.	Resultados experimentales .....	80
5.4.4.	Conclusiones.....	86
5.5.	REUTERS: 101 clases distribuidas jerárquicamente. ....	86
5.5.1.	Procedimiento experimental.....	88
5.5.2.	Características de los Modelos Evaluados.....	93
5.5.3.	Resultados Experimentales.....	98
5.5.4.	Conclusiones.....	101
5.6.	ASISTENTUR: Un problema real. ....	102
5.6.1.	Procedimiento experimental.....	103
5.6.2.	Características de los Modelos Evaluados.....	104
5.6.3.	Resultados Experimentales.....	108
5.6.4.	Efecto del Ruido .....	118
5.6.5.	Conclusiones.....	120
5.7.	Resumen y Discusión .....	121
<b>6.</b>	<b>Conclusiones y Trabajos Futuros.....</b>	<b>123</b>
6.1.	Conclusiones.....	124
6.2.	Trabajos Futuros .....	126
<b>7.</b>	<b>Bibliografía.....</b>	<b>127</b>
<b>8.</b>	<b>Apéndice A: Synthetic: <math>P_{MÁXIMA}</math> vs <math>P_{BCE}</math>.....</b>	<b>137</b>
<b>9.</b>	<b>Apéndice B: Estadísticos para la Clasificación de Textos.....</b>	<b>145</b>
<b>10.</b>	<b>Apéndice C: LYRL2004: Resultados Experimentales .....</b>	<b>149</b>

# Índice de Figuras

FIGURA 2.1. CONJUNTO DE CLASIFICADORES INTEGRADOS EN PARALELO..	16
FIGURA 2.2. CONJUNTO DE CLASIFICADORES INTEGRADOS EN SERIE.....	18
FIGURA 2.3. CONJUNTO DE CLASIFICADORES INTEGRADOS JERÁRQUICAMENTE .....	19
FIGURA 4.1. MÓDULO BASE ASOCIADO A LA <i>I-ÉSIMA</i> CLASE. ....	31
FIGURA 4.2. CONSTRUCCIÓN DE LOS CLASIFICADORES BINARIOS. ....	32
FIGURA 4.3. CONSTRUCCIÓN DEL CLASIFICADOR COMPLEMENTARIO ASOCIADO A LA <i>I-ÉSIMA</i> CLASE .....	35
FIGURA 4.4. INTEGRACIÓN BINARIO-COMPLEMENTARIO: A) PARALELO; B) SERIE; C) JERÁRQUICA. ....	36
FIGURA 4.5. ESQUEMA DE CONSTRUCCIÓN DE CCE. ....	42
FIGURA 5.1. EJEMPLOS DE LA BASE DE DATOS SYNTHETIC. ....	46
FIGURA 5.2. TASA DE ERROR DE LOS DISTINTOS CLASIFICADORES EN FUNCIÓN DEL NÚMERO DE CICLOS DE APRENDIZAJE.....	48
FIGURA 5.3. SYNTHETIC: PORCENTAJE DE ERROR EN FUNCIÓN DE $\theta$ . ....	51
FIGURA 5.4. SYNTHETIC: PORCENTAJE DE ERROR PARA LA COMBINACIÓN JERÁRQUICA. ....	52
FIGURA 5.5. SYNTHETIC: PRECISIÓN MÁX. Y PRECISIÓN MEDIA DE LOS MÓDULOS BASE FRENTE A LA PRECISIÓN DE BCE. ....	53
FIGURA 5.6. SYNTHETIC: MEJORA DE BCE ( $P_{BCE}-P_{MEDIA}$ ) FRENTE $Q, \rho, \kappa, DES$ . ....	55
FIGURA 5.7. SYNTHETIC: PRECISIÓN MÁX./MEDIA DE LOS CLASIFICADORES BASE FRENTE A LA PRECISIÓN DE <i>BAGGING</i> . ....	56
FIGURA 5.8. SYNTHETIC: MEJORA DE LA PRECISIÓN DE <i>BAGGING</i> ( $P_{BAGGING}-P_{MEDIA}$ ) FRENTE $Q, \rho, \kappa, DES$ . ....	56
FIGURA 5.9. SYNTHETIC: PRECISIÓN MÁX.Y PRECISIÓN MEDIA DE LOS MÓDULOS BASE FRETE A LA PRECISIÓN DE CCE.....	58
FIGURA 5.10. PORCENTAJE DE ERROR AL EVALUAR BCE Y CCE SOBRE LA BASE DE DATOS SYNTHETIC. ....	59
FIGURA 5.11. SYNTHETIC: MEJORA DE LA PRECISIÓN DE CCE ( $P_{CCE}-P_{MEDIA}$ ) FRENTE $Q, P, K, DES$ . ....	60
FIGURA 5.12. EJEMPLOS DE LA BASE DE DATOS SEMEION. ....	60
FIGURA 5.13. SEMEION: COMBINACIÓN EN SERIE. PORCENTAJE DE ERROR EN FUNCIÓN DE $\theta$ . ....	61
FIGURA 5.14. SEMEION: PORCENTAJE DE ERROR PARA LA COMBINACIÓN JERÁRQUICA. ....	62
FIGURA 5.15. SEMEION: PRECISIÓN MÁX.Y PRECISIÓN MEDIA DE LOS MÓDULOS BASE FRENTE A LA PRECISIÓN DE BCE. ....	63
FIGURA 5.16. SEMEION: MEJORA DE BCE ( $P_{BCE}-P_{MEDIA}$ ) FRENTE A $Q, \rho, \kappa, DES$ . ....	64
FIGURA 5.17. SEMEION: PRECISIÓN DE CCE EN FUNCIÓN DEL NÚM. DE MÓDULOS INTEGRADOS EN LA ARQUITECTURA. ....	65
FIGURA 5.18. PORCENTAJE DE ERROR AL EVALUAR BCE Y CCE SOBRE LA BASE DE DATOS SEMEION. ....	66
FIGURA 5.19. SEMEION: PRECISIÓN MÁX. Y PRECISIÓN MEDIA DE LOS MÓDULOS BASE FRENTE A LA PRECISIÓN DE CCE ....	67
FIGURA 5.20. MEJORA DE CCE ( $P_{CCE}-P_{MEDIA}$ ) FRENTE A 4 $Q, \rho, \kappa, DES$ . ....	67
FIGURA 5.21. PORCENTAJE DE ERROR AL EVALUAR BCE Y CCE SOBRE LA BASE DE DATOS SEMEION DESBALANCEADA. ....	69
FIGURA 5.22. SEMEION DESBALANCEADO: TASA DE ACIERTO SOBRE LAS CLASES MINORITARIAS. ....	70
FIGURA 5.23. SEMEION DESBALANCEADO: HISTOGRAMA DE LA PRECISIÓN OBTENIDA CON DISTINTOS SISTEMAS .....	71
FIGURA 5.24. EJEMPLOS DE LA BASE DE DATOS MNIST. ....	71
FIGURA 5.25 TABLA DE CONTINGENCIA PARA EL TEST DE McNEMAR. ....	73
FIGURA 5.26. PRECISIÓN FRENTE A NÚM. DE CICLOS DE APRENDIZAJE PARA DISTINTAS ARQUITECTURAS. DE UNA ANN.....	75
FIGURA 5.27. MNIST: TASA DE ACIERTO DE BCE EN FUNCIÓN DE $\theta$ PARA EL ESQUEMA DE INTEGRACIÓN EN SERIE. ....	78
FIGURA 5.28. MNIST:TASA DE ACIERTO DE BCE EN FUNCIÓN DE $\theta_1$ Y $\theta_2$ PARA EL ESQUEMA DE INTEGRACIÓN JERÁRQUICO. ....	78
FIGURA 5.29. SELECCIÓN DE ATRIBUTOS: ESQUEMAS DE CONSTRUCCIÓN DE LOS CLASIFICADORES BASE. ....	83
FIGURA 5.30. RCV1-V2- EJEMPLOS DE ENTRENAMIENTO ASIGNADOS A CADA CATEGORÍA. ....	87
FIGURA 5.31. RCV1-V2- EJEMPLOS DE TEST ASIGNADOS A CADA CATEGORÍA. ....	87

FIGURA 5.32. ALGORITMO PARA ESTABLECER EL <i>RANKING</i> DE ATRIBUTOS CONFORME AL VALOR DE $V(t, c_j)$ .	91
FIGURA 5.33. CONJUNTO DE CLASIFICADORES INTEGRADOS JERÁRQUICAMENTE.	93
FIGURA 5.34. LYRL2004: CATEGORÍAS DE LA RAMA JERÁRQUICA QUE INCLUYE A <i>MCAT</i> Y A TODOS SUS DESCENDIENTES. ...	94
FIGURA 5.35. ASISTENTUR: EJEMPLO DE LOS NUEVE TIPOS DE SEÑALES DE TRÁFICO.	102
FIGURA 5.36. ASISTENTUR: ERROR EN FUNCIÓN DEL NÚMERO DE CLASIFICADORES DE <i>BAGGING</i> .	105
FIGURA 5.37. ASISTENTUR: TASA DE ACIERTO EN FUNCIÓN DE $\theta$ .	106
FIGURA 5.38. ASISTENTUR: TASA DE ACIERTO EN FUNCIÓN DE $\theta_1$ Y $\theta_2$ .	107
FIGURA 5.39. ASISTENTUR: ERROR EN FUNCIÓN DEL NÚMERO DE MÓDULOS BASE DE CCE.	108
FIGURA 5.40. EVALUACIÓN DEL RUIDO. ESQUEMA SEGUIDO EN LA CONSTRUCCIÓN DE LOS CLASIFICADORES.	118
FIGURA B.1. CURVA NORMAL TIPIFICADA.	146



# Índice de Tablas

TABLA 4.1. ESQUEMA DE CODIFICACIÓN DE CCE PARA UN PROBLEMA DE 6 CLASES. ....	38
TABLA 4.2. NÚM. MÁXIMO DE MÓDULOS BASE PARA LOS SISTEMAS BCE Y CCE EN FUNCIÓN DEL NÚM. DE CLASES. ....	39
TABLA 4.3. MATRIZ DE DESCOMPOSICIÓN DE CLASES DE CCE PARA UN PROBLEMA DE 6 CLASES.. ....	40
TABLA 5.1. PRINCIPALES CARACTERÍSTICAS DE LOS DOMINIOS EVALUADOS.....	46
TABLA 5.2. SYNTHETIC: PORCENTAJE DE ERROR EN FUNCIÓN DE $\theta$ . ....	50
TABLA 5.3. SYNTHETIC: ANÁLISIS DE VARIANZA DE UN FACTOR. ....	51
TABLA 5.4. DEFINICIÓN DE ALGUNAS MEDIDAS DE DIVERSIDAD.....	54
TABLA 5.5. SYNTHETIC: ERROR MEDIO OBTENIDO AL REALIZAR 30 ITERACIONES DEL PROCESO DE VALIDACIÓN CRUZADA. ....	57
TABLA 5.6. SYNTHETIC: COMPARACIÓN ESTADÍSTICA (ESTADÍSTICO T PAREADO) DE BCE Y CCE.....	59
TABLA 5.7. SEMEION: ANÁLISIS DE VARIANZA DE UN FACTOR. ....	62
TABLA 5.8. SEMEION. ANÁLISIS DE VARIANZA: COMBINACIÓN JERÁRQUICA.....	63
TABLA 5.9. SEMEION: COMPARACIÓN ESTADÍSTICA (PRUEBA T PAREADA) DE BCE Y CCE.....	66
TABLA 5.10. COEFICIENTE DE CORRELACIÓN DE PEARSON Y COEFICIENTE DE CORRELACIÓN DE SPEARMAN ENTRE LA MEJORA DEL CONJUNTO ( $P_{CJTO}-P_{MEDIA}$ ) Y LAS DISTINTAS MEDIDAS DE DIVERSIDAD ( $Q, \rho, \kappa, DES$ ).....	68
TABLA 5.11. DISTRIBUCIÓN DE EJEMPLOS EN LA BASE DE DATOS MNIST. ....	72
TABLA 5.12. DEFINICIÓN DE DOS DE LAS MEDIDAS DE DIVERSIDAD USADAS.....	73
TABLA 5.13. CÓDIGO CORRECTOR DE ERRORES PARA EL RECONOCIMIENTO DE DÍGITOS MANUSCRITOS. ....	77
TABLA 5.14. MATRIZ DE DESCOMPOSICIÓN DE CLASES DE CCE PARA EL DOMINIO MNIST.. ....	79
TABLA 5.15. RESULTADOS EXPERIMENTALES OBTENIDOS SOBRE MNIST.....	80
TABLA 5.16. MNIST: ERRORES COMETIDOS POR CADA MODELO Y VALORES DEL TEST DE McNEMAR.. ....	81
TABLA 5.17. MNIST: NÚMERO DE VECES EN LAS QUE UN SISTEMA GANA/EMPATA/PIERDE FRENTE A LOS DEMÁS. ....	81
TABLA 5.18. MNIST: TIEMPOS DE CLASIFICACIÓN Y APRENDIZAJE DE CADA CLASIFICADOR BASE.....	82
TABLA 5.19. MNIST: ERRORES OBTENIDOS AL EVALUAR EL MÉTODO DE SELECCIÓN DE CARACTERÍSTICAS SOBRE OAA, BAGGING Y ECOC.....	83
TABLA 5.20. MNIST: RESULTADOS EXPERIMENTALES OBTENIDOS CUANDO LOS MODELOS DE CLASIFICACIÓN SE CONSTRUYEN A PARTIR DE EJEMPLOS SOBRE LOS QUE SE HA REALIZADO UN PROCESO DE SELECCIÓN DE ATRIBUTOS.. ....	84
TABLA 5.21. MNIST: ERRORES COMETIDOS POR CADA MODELO Y VALORES DEL TEST DE McNEMAR. ....	84
TABLA 5.22. MNIST: NÚMERO DE VECES EN LAS QUE, SEGÚN EL TEST DE McNEMAR, UN SISTEMA GANA/EMPATA/PIERDE FRENTE A LOS DEMÁS. ....	85
TABLA 5.23 VALORES DE DIVERSIDAD Y PRECISIÓN PARA BAGGING, BCE Y CCE SOBRE MNIST.....	85
TABLA 5.24. REUTERS: CODIFICACIÓN DE CLASES DE BCE ASOCIADA AL NIVEL 1. ....	97
TABLA 5.25. REUTERS: MATRIZ DE DESCOMPOSICIÓN DE CÓDIGOS PARA LOS MÓDULOS DE CCE DE NIVEL 1.....	97
TABLA 5.26. REUTERS: VALORES DE $\pi, \rho$ Y $F_1$ OBTENIDOS AL EVALUAR BCE Y CCE SOBRE LAS CLASES NIVEL 1. ....	98
TABLA 5.27. VALORES DE $\pi, \rho$ , Y $F_1$ A NIVEL MICRO ( $\mu$ ) Y MACRO (M) PARA LOS DISTINTOS MODELOS EVALUADOS. ....	99
TABLA 5.28. RESULTADOS OBTENIDOS AL APLICAR EL ESTADÍSTICO S-TEST. ....	100
TABLA 5.29. RESULTADOS OBTENIDOS AL APLICAR EL ESTADÍSTICO S-TEST. ....	100
TABLA 5.30. RESULTADOS OBTENIDOS AL APLICAR EL ESTADÍSTICO T-TEST.....	100
TABLA 5.31. RESULTADOS OBTENIDOS AL APLICAR EL ESTADÍSTICO P-TEST SOBRE LOS VALORES DE PRECISIÓN ( $\pi$ ).....	100
TABLA 5.32. RESULTADOS OBTENIDOS AL APLICAR EL ESTADÍSTICO P-TEST SOBRE LOS VALORES DE SENSIBILIDAD ( $\rho$ ).....	101

TABLA 5.33. CONCLUSIONES EXTRAÍDAS TRAS LA APLICACIÓN DE LOS DISTINTOS ESTADÍSTICOS.....	101
TABLA 5.34. CÓDIGO CORRECTOR DE ERRORES PARA EL RECONOCIMIENTO DE SEÑALES DE TRÁFICO..	106
TABLA 5.35. ASISTENTUR: ERRORES COMETIDO POR UN MODELOS DE CLASIFICACIÓN.....	109
TABLA 5.36. ASISTENTUR: RESULTADOS EXPERIMENTALES (I) .....	110
TABLA 5.37. ASISTENTUR: NÚM. DE ERRORES COMETIDO POR LOS DISTINTOS MODELOS .	110
TABLA 5.38. ASISTENTUR: COMPARACIÓN ESTADÍSTICAS DE LOS MODELOS IMPLEMENTADOS..	111
TABLA 5.39. ASISTENTUR: TIEMPOS DE CLASIFICACIÓN Y APRENDIZAJE DE CADA CLASIFICADOR BASE .....	111
TABLA 5.40. ASISTENTUR: ERRORES OBTENIDOS AL EVALUAR EL MÉTODO DE SELECCIÓN DE CARACTERÍSTICAS SOBRE OAA, BAGGING Y ECOC.....	112
TABLA 5.41. ASISTENTUR: RESULTADOS EXPERIMENTALES (II) .....	113
TABLA 5.42. ASISTENTUR: NÚM. DE ERRORES COMETIDO POR LOS DISTINTOS MODELOS CUANDO ÉSTOS SE COMBINAN CON UN PROCESO DE SELECCIÓN DE ATRIBUTOS. ....	113
TABLA 5.43. ASISTENTUR: COMPARACIÓN ESTADÍSTICAS DE LOS MODELOS IMPLEMENTADOS. ....	114
TABLA 5.44. ASISTENTUR: VALORES DE DIVERSIDAD Y PRECISIÓN PARA BAGGING, BCE Y CCE. ....	115
TABLA 5.45. COEFICIENTE DE NO DETERMINACIÓN ( $1-R^2$ ) ENTRE LA DIVERSIDAD Y LA MEJORA DEL CONJUNTO .....	117
TABLA 5.46. ASISTENTUR: PRECISIÓN POR CLASES Y PRECISIÓN GLOBAL DADA POR LOS DISTINTOS CLASIFICADORES CUANDO SE EVALÚA SU ROBUSTEZ FRENTE AL RUIDO..	119
TABLA 5.47. ERRORES COMETIDOS POR LOS DISTINTOS MODELOS CUANDO SE EVALÚA SU ROBUSTEZ FRENTE AL RUIDO. ....	119
TABLA 5.48. COMPARACIÓN DE LOS MODELOS IMPLEMENTADOS CUANDO SE EVALÚA SU ROBUSTEZ FRENTE AL RUIDO.....	120
TABLA 5.49. RESUMEN DE LOS DOMINIOS ESTUDIADOS. ....	1201
TABLA A.1. SYNTHETC. $P_{MÁX}$ , $P_{MÍN}$ , $P_{MEDIA}$ Y $P_{BCE}$ PARA LOS 600 MODELOS CONSTRUIDOS. ....	137
TABLA B.1. VALOR NORMAL TIPIFICADO PARA $\alpha=0,05$ Y $\alpha=0,01$ . ....	146
TABLA C.1. VALORES OBTENIDOS AL EVALUAR EL MODELO1 (ANN) SOBRE LA BASE DE DATOS LYRL2004. ....	150
TABLA C.2. VALORES OBTENIDOS AL EVALUAR EL MODELO2 (OAA) SOBRE LA BASE DE DATOS LYRL2004. ....	152
TABLA C.3. VALORES OBTENIDOS AL EVALUAR EL MODELO3 (BCE) SOBRE LA BASE DE DATOS LYRL2004. ....	154
TABLA C.4. VALORES OBTENIDOS AL EVALUAR EL MODELO4 (CCE) SOBRE LA BASE DE DATOS LYRL2004. ....	156
TABLA C.5. VALORES OBTENIDOS AL EVALUAR <i>knn</i> OPTIMIZADO PARA MICRO-VALORES SOBRE LYRL2004 .....	158
TABLA C.6. VALORES OBTENIDOS AL EVALUAR <i>knn</i> OPTIMIZADO PARA MACRO-VALORES SOBRE LYRL2004 .....	160
TABLA C.7. VALORES OBTENIDOS AL EVALUAR <i>Rocchio</i> SOBRE LA BASE DE DATOS LYRL2004. ....	162
TABLA C.8. VALORES OBTENIDOS AL EVALUAR <i>SVM1</i> OPTIMIZADO PARA MICRO-MEDIDAS SOBRE LYRL2004.....	164
TABLA C.9. VALORES OBTENIDOS AL EVALUAR <i>SVM1</i> OPTIMIZADO PARA MACRO-MEDIDAS SOBRE LYRL2004.....	166

# Capítulo 1

## Introducción

Un clasificador es un sistema que permite asignar categorías o clases a los datos que se le presentan. Esta tarea, que a priori puede parecer sencilla, resulta inviable si el clasificador no cuenta con el conocimiento necesario para poder realizarla. En otras palabras, para reconocer y, por tanto, clasificar los datos u objetos pertenecientes a una determinada categoría primero es necesario aprender, a partir de la experiencia, la *esencia* de dicha categoría. Aunque es factible adquirir este conocimiento llevando a la práctica distintas formas de aprendizaje (deducción, analogía, memorización), la forma más habitual de adquirirlo es *inducirlo* a partir de ejemplos o datos previamente clasificados. En este caso, el proceso de aprendizaje se denomina *aprendizaje supervisado*. Según señala Pozo, [Pozo, 1989], ésta es la forma en la que los niños adquieren la mayor parte de sus conceptos, es decir, observando datos cuya categoría es conocida y construyendo hipótesis o modelos que permiten identificar de forma fehaciente los datos sobre los que se ha aprendido. Una vez generada la hipótesis, es decir, aprendida la esencia del concepto, ésta se utiliza para determinar la clase o categoría a la que pertenecen los nuevos datos que se puedan presentar.

Las tareas de clasificación están presentes en una amplia variedad de dominios (diagnóstico médico o técnico, detección de fraudes, detección de errores en un proceso de fabricación, reconocimiento de formas o caracteres, etc.) en los que, en general, se manejan cantidades ingentes de información. Por ello, desde mediados del siglo XX, gracias al desarrollo tecnológico e impulsado por el afán de construir sistemas informáticos que actúen racionalmente, se ha promovido el diseño de métodos y modelos que automaticen las tareas de clasificación. Es decir, se ha promovido la construcción de sistemas capaces, por un lado, de inducir conocimiento a partir de ejemplos o datos y, por otro, de usar el conocimiento adquirido para clasificar de forma eficiente nuevos datos que se puedan presentar.

Los primeros programas de aprendizaje -década de los 50- estuvieron basados en las teorías conexionistas del comportamiento humano, es decir, se sustentaron en la idea de

que los procesos cognitivos, entre los que se incluyen el aprendizaje y el razonamiento, emergen de la interacción neuronal. Por tanto, para simular computacionalmente estos procesos se intentó emular la estructura fisiológica de las neuronas y sus interconexiones es decir, se intentó mimetizar desde un punto de vista funcional las *redes de neuronas* [Minsky, 1954]. Para los sistemas basados en este paradigma, aprender significa modificar o ajustar los pesos de los enlaces existentes entre las distintas neuronas para así conseguir determinados patrones de activación [Rumelhart & McClelland, 1988]. Esta forma de codificar el conocimiento ha permitido construir clasificadores especialmente útiles en aquellos casos en los que los ejemplos son incompletos, ruidosos, aparentemente contradictorios o que siguen una distribución no lineal.

Históricamente, la imposibilidad de interpretar y extrapolar el conocimiento adquirido por una red de neuronas y el recelo de los profesionales del dominio a usar las conclusiones dadas por un sistema sin conocer los motivos por los que se han tomado hizo que el interés por los denominados *sistemas simbólicos* fuera en aumento. En otras palabras, las limitaciones propias de los modelos conexionistas han hecho que muchas de las investigaciones concernientes a tareas de clasificación se centren en desarrollar sistemas en los que el conocimiento adquirido y las conclusiones derivadas de él sean fáciles de entender e interpretar. Este es el caso de los árboles de decisión (ID3 [Quinlan, 1986], C4.5 [Quinlan, 1993]) y de los sistemas basados en reglas (AQ [Michalski, 1983], PART [Frank & Witten, 1998]), en los que el conocimiento adquirido está expresado en un lenguaje inteligible por el ser humano.

Junto a los paradigmas conexionista y simbólico, en los que el objetivo es encontrar una descripción general de los ejemplos suministrados, existe un tercer paradigma que intenta resolver las tareas de clasificación en función de la posición que un objeto (dato) ocupa en el espacio y de la distancia que le separa de otros objetos. Este paradigma, denominado *aproximación estadística*, sostiene que las fronteras entre clases están delimitadas por las densidades de probabilidad dependientes de cada clase, densidades éstas que pueden obtenerse a partir de un conjunto de datos de entrenamiento. En función de si la forma de estas densidades es o no conocida a priori, los métodos estadísticos se dividen en *paramétricos* o *no paramétricos*. En los métodos paramétricos el problema se reduce a estimar, a partir de los datos de entrenamiento, los parámetros de la distribución, supuestamente conocida, que sigue las muestras de cada clase. Los modelos Bayesianos [Mitchell, 1997] son los métodos más usados para estimar el valor de estos parámetros. Por el contrario, en las aproximaciones no paramétricas la función de densidad de cada clase se determina en función del número de ejemplos existentes en cada región del espacio. Los *k-vecinos* más cercanos (*K-Nearest Neighbor*) [Martin, 1995] es, quizás, el algoritmo más representativo de este paradigma.

En la década de los 90 las investigaciones concernientes al Aprendizaje Automático y, en particular las relativas a las tareas de clasificación, se ven impulsadas por la idea de usar sistemas *multiestratégicos* o *híbridos*, es decir, sistemas en los que se combinan distintos métodos de inferencia y/o técnicas de aprendizaje [Michalski, 1994]. Ejemplos de este tipo de sistemas son EITHER [Mooney & Ourston, 1994] o CLINT [de Raedt & Bruynooghe, 1994] cuyo objetivo es refinar una teoría imperfecta usando, para ello, inferencias inductivas, deductivas o analógicas. Otros sistemas englobados en esta taxonomía son, por ejemplo, el presentado en [Burns & Danyluk, 2000], en el que se

combinan algoritmos genéticos y redes de neuronas para realizar selección de características, o el propuesto en [Gutiérrez et al., 2005], donde se sugiere automatizar el diseño de la topología de una red de neuronas haciendo uso de técnicas evolutivas. Otro tipo de sistemas híbridos son aquellos en los que la tarea de clasificación se resuelve gracias a la intervención de una colección de clasificadores de distinta naturaleza. Estos sistemas reciben, entre otros, el nombre de *conjuntos de clasificadores* y se han convertido en una de las principales líneas de investigación de las dos últimas décadas.

Bajo el denominador de *conjuntos de clasificadores* se engloban un gran número de algoritmos (*Boosting* [Schapire, 1990], *Stacking* [Wolper, 1992], *ECOC* [Dietterich & Bakiri, 1995], *Bagging* [Breiman, 1996], etc.) cuyo objetivo es la construcción de un clasificador robusto haciendo uso de clasificadores más simples denominados *clasificadores base*. Atendiendo a esta definición, los conjuntos de clasificadores pueden ser catalogados en dos grandes grupos:

- Por un lado están aquellos sistemas en los que los clasificadores base son redundantes, es decir, todos ellos resuelven la misma tarea. La idea que subyace a estos sistemas es que la aplicación de diferentes métodos de aprendizaje a los mismos datos o la aplicación de un mismo método a conjuntos de datos distintos conduce a resultados no siempre equivalentes, es decir, produce modelos distintos [Somerén, 2001]. Por tanto, es posible que un modelo cuente con conocimiento que otros modelos no han sido capaces de encontrar y, por tanto, que al combinar sus decisiones se obtengan resultados más precisos que los que se obtendrían usando un único modelo. En función de la técnica usada para integrar los resultados ofrecidos por los distintos módulos, esta arquitectura tiene dos variantes: *selección y fusión* [Saitta, 2001]. Cuando se usa selección, el resultado del proceso de clasificación se corresponde con el ofrecido por un único clasificador base. Por el contrario, cuando se usa fusión, el resultado de la clasificación es función de los resultados dados por los distintos clasificadores base.
- Por otro lado, están aquellos sistemas en los que la tarea de clasificación se descompone en varias subtarear cada una de las cuales es resuelta por un clasificador base. En este tipo de sistemas, denominados por algunos autores *sistemas modulares* [Sharkey, 1999], la solución completa al problema de clasificación requiere la contribución de todos los clasificadores base. Entre los modelos que se ajustan a esta variante estarían aquellos que resuelven los problemas multiclase descomponiéndolos en problemas binarios o aquellos en los que cada clasificador aprende de datos provenientes de fuentes (p.e. sensores) distintas.

En las últimas décadas, y dentro del marco del *Aprendizaje Automático*, el *Reconocimiento de Patrones* y la *Estadística*, el estudio de los modelos de clasificación es una de las áreas que más interés ha suscitado. Sin embargo, y a pesar del gran número de estudios acaecidos, a día de hoy no se puede aseverar que exista el *clasificador universal*. Es decir, dado que la calidad de un modelo depende considerablemente de las características específicas del dominio sobre el que se aplica, la búsqueda del clasificador perfecto en *todos los dominios y contextos*, sigue siendo

una utopía [Ranawana, 2006]. Además, en ciertas ocasiones, la adecuación de un modelo de clasificación a un determinado problema puede verse condicionada por factores colaterales a la tarea de clasificación propiamente dicha. Así, el mejor clasificador puede ser el más preciso pero también, el más simple, el más eficiente, el que se ajusta a un determinado paradigma (conexionista, simbólico o estadístico) o el que mejor satisface la combinación de éstos u otros criterios.

Un caso particular de este tipo de problemas, y punto de partida para el desarrollo de esta Tesis [Sesmero, et al., 2007a; Sesmero, et al., 2007b] es la clasificación e identificación de señales de tráfico. En este caso específico, la necesidad de integrar el módulo de clasificación en un sistema de ayuda a la conducción capaz de trabajar en tiempo real impone ciertas restricciones de diseño que dificultan su construcción. En concreto, para lograr que el modelo *software* desarrollado pueda integrarse en un sistema *hardware* de ayuda a la conducción, se impone como condición que el algoritmo de clasificación esté basado en el paradigma conexionista y que tanto el coste como la complejidad computacional no sobrepasen ciertas cotas.

Tras analizar desde diversas perspectivas el comportamiento que en este y otros dominios similares presentan algunos de los modelos recogidos en la Bibliografía, se han desarrollado dos nuevos conjuntos de clasificadores basados en redes de neuronas. Estos nuevos modelos han sido concebidos explícitamente para la resolución de problemas *multiclase* en los que los ejemplos están descritos por un elevado número de atributos. Partiendo de estas dos premisas se pretende garantizar que tanto la complejidad como el coste computacional inherentes a este tipo de clasificadores sean moderados y, en consecuencia, que puedan ser utilizados para la clasificación en tiempo real.

Hasta aquí se ha comentado a grandes trazos qué se entiende por clasificador, qué paradigmas de aprendizaje se pueden usar en su construcción y alguno de los objetivos y limitaciones que marcan el desarrollo de este trabajo. En el capítulo 2 se profundiza en los conjuntos de clasificadores, presentando el estado del arte de estos sistemas, sus variantes y características más relevantes. Este estudio sirve para fijar los objetivos de esta Tesis Doctoral que están recogidos en el capítulo 3. A continuación, en el capítulo 4 se describe y analiza la arquitectura de los conjuntos de clasificadores propuestos. Seguidamente, en el capítulo 5 se detallan y examinan los resultados empíricos obtenidos al evaluar cada una de las propuestas realizadas sobre distintos dominios y se indica el método experimental seguido en cada una de las evaluaciones. Finalmente, en el Capítulo 6 se recogen las principales conclusiones obtenidas tras la elaboración de este trabajo y las líneas futuras de investigación.

## Capítulo 2

### Estado del Arte

El objetivo de este capítulo es brindar una visión general del área en la que se enmarca esta Tesis Doctoral. A manera de introducción, en la sección 2.1 se desarrolla el concepto de *conjunto de clasificadores*. Posteriormente, en las secciones 2.2 y 2.3 se detallan las fases requeridas para construir un conjunto de clasificadores. Por último, en la sección 2.4 se muestran algunos estudios comparativos y las conclusiones derivadas de ellos.

#### 2.1. Conjunto de Clasificadores

Según Dietterich, [Dietterich, 2000a] un conjunto de clasificadores (*ensemble of classifiers*) es un grupo de clasificadores cuyas decisiones individuales se combinan con el objetivo de obtener un clasificador más preciso que cualquiera de sus miembros.

Al igual que otras muchas ideas aplicadas en el campo de la Inteligencia Artificial, los conjuntos de clasificadores responden a un intento de emular el comportamiento humano. En concreto, estos sistemas intentan, en líneas generales, reproducir la actuación del ser humano cuando éste se enfrenta a la toma de decisiones de relativa importancia. Así, por ejemplo, es habitual buscar la opinión de diferentes doctores antes de someternos a una operación, leer distintas opiniones antes de comprar un determinado producto o consultar a los miembros de la junta directiva de una empresa antes de tomar una decisión de carácter financiero. Es decir, antes de tomar una decisión importante parece razonable conocer la opinión de distintos expertos y, en función de estas opiniones, tomar la decisión que parece más adecuada. Para ello, es habitual no solo valorar y ponderar las distintas aportaciones sino, ante una falta de unanimidad, decidir cuál o cuáles de ellas son las más acertadas en cada caso [Polikar, 2012].

La extrapolación de esta proposición al campo del Aprendizaje Automático deriva en la construcción de sistemas compuestos por varios clasificadores en los que la decisión

final se toma de forma colectiva. Es decir, en la construcción de conjuntos de clasificadores.

Los conjuntos de clasificadores se construyen en dos fases [Polikar, 2006]: la fase de *generación* y la fase de *integración*. El objetivo de la fase de generación es construir, a partir de los datos de entrenamiento, los clasificadores individuales que formarán parte del conjunto. Estos clasificadores se denominan clasificadores base. En la fase de integración, las hipótesis generadas por los clasificadores base se combinan con el fin de obtener una hipótesis más precisa que cualquiera de las hipótesis individuales.

A continuación se describen en detalle cada una de estas fases.

## 2.2. Fase de generación de los clasificadores base

Para que un conjunto de clasificadores mejore la precisión de cualquiera de los miembros que lo componen, se requiere que éstos sean precisos y diversos [Hansen & Salamon, 1990]. Un clasificador se dice que es preciso si el error cometido al clasificar nuevos ejemplos es menor que el que se cometería asignando una clase de forma aleatoria. Por otro lado, los clasificadores se consideran diversos si las decisiones erróneas se producen sobre ejemplos distintos.

Requerir que los clasificadores base sean precisos es una condición fácil de justificar y verificar. Partiendo de hipótesis imprecisas, difícilmente se conseguirá una hipótesis precisa. Además, una forma sencilla y bastante aceptada para determinar la precisión de un clasificador es calcular el porcentaje de ejemplos de test correctamente clasificados. Si el valor obtenido es mayor que el que se obtendría clasificando los ejemplos de forma aleatoria, el clasificador se puede considerar preciso.

Para clarificar la necesidad de requerir clasificadores base diversos se considerará un ejemplo en el que existen tres clasificadores,  $h_1$ ,  $h_2$  y  $h_3$  y un ejemplo a clasificar,  $x$ . Si los clasificadores no son diversos, es decir, cometen los mismos errores, y el ejemplo es clasificado erróneamente por uno de ellos, también será clasificado erróneamente por los otros dos. Por tanto, la decisión final del conjunto clasificador será errónea. Por el contrario, si los clasificadores base son diversos, cuando la predicción dada por  $h_1$  sea errónea, las dadas por  $h_2$  y  $h_3$  serán correctas y, por tanto, si todas ellas tienen igual relevancia, al combinarlas, la predicción global del sistema será correcta.

Aunque está demostrado que una condición necesaria para obtener un conjunto de clasificadores más preciso que los clasificadores base que lo integran es que éstos sean diversos, encontrar un grupo de clasificadores diversos con los que se garantice que el conjunto final es preciso no es una tarea trivial. La principal razón está en que, al contrario de lo que sucede con la precisión, los investigadores de este campo no han llegado a un consenso sobre cómo cuantificar esta magnitud ni cómo relacionarla con la precisión del conjunto. Es decir, hay varios parámetros que permiten estimar la diversidad de los clasificadores base pero la relación entre tales parámetros y la precisión del conjunto de clasificadores es aún una cuestión abierta [Kuncheva & Whitaker, 2003; Polikar, 2012].

No obstante, y dado que para lograr un buen conjunto de clasificadores es necesario partir de clasificadores base diversos, a continuación se exponen las técnicas más



aceptadas que permiten obtener clasificadores que, en mayor o menor grado, satisfacen este requisito.

### 2.2.1. Generación de clasificadores diversos

Las técnicas usadas para generar clasificadores diversos se sustentan en el hecho de que la hipótesis asociada a un clasificador depende del algoritmo de aprendizaje y de los ejemplos usados en su construcción. Por tanto, variar el algoritmo de aprendizaje y/o el conjunto de datos de entrenamiento usado en la construcción de una serie de clasificadores puede derivar en que los clasificadores resultantes se comporten de manera distinta sobre un determinado conjunto de datos, es decir, que sean diversos.

Experimentalmente se ha demostrado que una forma de conseguir que los miembros del conjunto clasificador sean diversos es construirlos usando distintos algoritmos de aprendizaje, es decir, basando la construcción del conjunto clasificador en sistemas heterogéneos en los que, por ejemplo, se pueden combinar árboles de decisión, redes de neuronas o modelos bayesianos [Wolpert, 1992; Bahler & Navarro, 2000].

Otra técnica efectiva para obtener clasificadores diversos es la incorporación de aleatoriedad en el algoritmo de aprendizaje. Por ejemplo, en [Kolen & Pollack, 1991] se muestra como entrenando una red de neuronas con el mismo conjunto de entrenamiento pero con diferentes pesos iniciales, se pueden obtener clasificadores que, ante un mismo patrón de entrada, generan respuestas distintas. Otro ejemplo de este tipo de técnicas es *Randomization* [Dietterich, 2000b] caracterizado por generar árboles de decisión en los que el criterio usado para expandir un nodo no es fijo, sino que se elige al azar de entre un conjunto de posibles criterios.

Por otro lado, las técnicas desarrolladas para generar diversidad variando el conjunto de datos de entrenamiento pueden ser clasificadas en tres grandes grupos [Dietterich, 1997]: manipulación del conjunto de entrenamiento, manipulación de los atributos de entrada y manipulación de las etiquetas que describen la clase. A continuación se analizan cada una de estas técnicas.

- Manipulación del conjunto de entrenamiento. Una forma de generar clasificadores diversos es construir cada clasificador basándose en un subconjunto de entrenamiento distinto. Si los ejemplos usados en la construcción del clasificador son diferentes, las hipótesis obtenidas serán también diferentes.

Dentro de las técnicas que manipulan el conjunto de entrenamiento con el objetivo de generar clasificadores diversos destacan *Bagging* [Breiman, 1996], los *comités de validación cruzada* (en inglés, *cross-validated committees*) [Parmanto et al., 1996] y *Boosting* [Schapire 1990].

Así, *Bagging*, a partir del conjunto de ejemplos original genera un grupo de subconjuntos en los se mantiene la cardinalidad inicial pero en los que, con respecto al conjunto original, algunas instancias están repetidas y otras omitidas.

Otra forma de generar distintos subconjuntos de entrenamiento es dividir el conjunto de datos inicial en  $K$  particiones disjuntas de igual tamaño y, posteriormente, generar  $K$  subconjuntos distintos compuestos, cada uno de ellos,

por todos los elementos contenidos en  $K-1$  de estas particiones. Este proceso coincide con el usado en validación cruzada por lo que los métodos que lo aplican se denominan *comités de validación cruzada*.

Finalmente, *Boosting*, a diferencia de los dos métodos anteriormente expuestos, genera los conjuntos de entrenamiento de manera secuencial, es decir, cada subconjunto depende de los generados anteriormente. Esta dependencia está relacionada con la precisión del clasificador al que cada subconjunto está asociado pues, en esencia, la probabilidad de que un ejemplo pase a formar parte de un nuevo subconjunto de entrenamiento depende de si los clasificadores construidos anteriormente lo clasificaron o no correctamente.

- Manipulación de los atributos de entrada. Otra técnica para generar los clasificadores base de un conjunto de clasificadores es modificar cuantitativa y/o cualitativamente el conjunto de atributos con los que se definen los patrones. La modificación cuantitativa permite reducir la cantidad de atributos en cada uno de los subconjuntos de entrenamiento. Esta reducción puede realizarse basándose en una selección aleatoria [Ho, 1998; Bryll et al., 2003], aplicando distintos algoritmos de selección de características [Blum & Langley, 1997; Tsymbal, et al., 2006] o aplicando conocimiento dependiente del dominio (p.e., agrupando los atributos en función de la fuente de la que proceden). Las modificaciones cualitativas están vinculadas a procesos de inducción constructiva [Zheng, 1996] y, por tanto, conllevan un cambio en el espacio de representación de los atributos que describen los ejemplos (cambio en los valores que pueden tomar o generación de nuevos atributos). En esta última línea estarían los procesos de transformación no lineal propuestos en [Sharkey & Sharkey, 1997] en los que el cambio en la descripción de los ejemplos se consigue i) entrenando una red de neuronas de una capa oculta en la que tanto las entradas como las salidas coinciden con la descripción inicial de los ejemplos y tomando la activación de las neuronas de la capa oculta como el nuevo espacio de representación, o ii) transformando los datos con ayuda de una red de neuronas arbitraria (no entrenada) y considerando que el valor de activación de las neuronas de la capa de salida se corresponde con el nuevo espacio de representación.
- Manipulación de las salidas. Una tercera técnica para generar clasificadores diversos es modificar la clase asociada a las instancias. Estos métodos tienen su mayor utilidad cuando los datos de entrenamiento están asociados a un número grande de clases. El procedimiento más habitual consiste en descomponer el problema *multiclase* en subproblemas binarios, resolver estos problemas y combinar las soluciones encontradas y así dar solución al problema inicial.

En función del esquema usado en la descomposición del problema, estos sistemas se pueden subdividir en las siguientes categorías [Ou & Murphey, 2007]:

- Sistemas basados en una arquitectura *uno contra todos* (OAA, del inglés, *One Against All*) [Anand et al., 1992]. En esta arquitectura cada clasificador binario se especializa en separar los ejemplos pertenecientes a una clase respecto de los ejemplos pertenecientes a cualquiera de las clases restantes.

- Sistemas basados en una arquitectura *uno contra uno* (OAO, del inglés, *One Against One*) [Hastie & Tibshirani, 1998]. En este caso, cada clasificador binario tiene como objetivo distinguir los ejemplos de una determinada clase respecto de los pertenecientes a otra clase dada.
- Sistemas basados en una arquitectura *p contra q*. Al aplicar esta metodología, cada clasificador binario se especializa en distinguir los ejemplos asociados a  $P$  clases respecto de los asociados a  $Q$  clases ( $P, Q \geq 1$ ;  $P+Q \leq N^{\circ}$  total del clases).

Un método representativo de las técnicas basadas en la manipulación de las salidas y, en particular, de la arquitectura *p contra q* es ECOC (*Error correcting output code*) [Dietterich & Bakiri, 1995]. En este método los ejemplos de entrada son re-etiquetados como pertenecientes a dos clases, es decir, el conjunto de clases inicial  $C=\{c_1, c_2, \dots, c_k\}$  es dividido aleatoriamente en dos subconjuntos  $C_i^+$  y  $C_i^-$  de forma que los ejemplos asociados a las clases contenidas en  $C_i^+$  son re-etiquetados con 1 y las instancias pertenecientes a cualquier clase contenida en  $C_i^-$  son re-etiquetadas con 0. Al repetir este proceso  $L$  veces se obtienen  $L$  conjuntos de entrenamiento distintos y, en consecuencia,  $L$  clasificadores distintos.

Una vez creados todos los clasificadores base, la reconstrucción del problema inicial implica admitir que una instancia clasificada con un 1 por el *i-ésimo* clasificador podrá pertenecer a cualquiera de las clases contenidas en  $C_i^+$  y, por tanto, cada una de ellas recibirá un voto. Por el contrario, si la salida del *i-ésimo* clasificador es 0, la instancia podrá pertenecer a cualquiera de las clases contenidas en  $C_i^-$ , por lo que cada clase en  $C_i^-$  recibirá un voto. Una vez que se conoce la decisión dada por los  $L$  clasificadores, la etiqueta que se le asigna a la instancia dependerá del número de votos que ha recibido cada una de las clases.

## 2.2.2. Medidas de diversidad

Reiterando lo ya señalado, una de las condiciones necesarias para obtener un buen conjunto de clasificadores es que los clasificadores base que lo integran sean diversos. No obstante, y a pesar de su relevancia, a día de hoy no existe una definición formal de este parámetro y, en consecuencia, no existe una expresión estándar con la que cuantificarla. Sin embargo, en la literatura, es posible encontrar medidas estadísticas y matemáticas cuyo valor se puede considerar un indicativo del grado de diversidad existente entre los miembros del conjunto.

Dado que determinar el grado de diversidad entre dos clasificadores es intuitivamente más sencillo que estimar el grado de diversidad asociado a un grupo de clasificadores, estas medidas se suelen catalogar en dos grupos: medidas duales (*pairwise*) y medidas no duales (*no pairwise*).

## Medidas duales

Estas medidas intentan establecer la diversidad existente entre las decisiones asociadas a dos clasificadores base. De ahí que cuando el conjunto está formado por 3 o más clasificadores base la estimación de la diversidad esté dada por el promedio de las medidas sobre todos los pares de clasificadores base.

Admitiendo que dos clasificadores se consideran diversos si las decisiones erróneas se producen sobre ejemplos distintos, parece claro que el grado de diversidad entre dos clasificadores,  $C_i$  y  $C_j$ , ha de ser función de:

$N^{11}$ : Número de ejemplos clasificados correctamente por ambos clasificadores.

$N^{10}$ : Número de ejemplos clasificados correctamente por el clasificador  $C_i$  pero erróneamente por el clasificador  $C_j$ .

$N^{01}$ : Número de ejemplos clasificados correctamente por el clasificador  $C_j$  pero erróneamente por el clasificador  $C_i$ .

$N^{00}$ : Número de ejemplos clasificados erróneamente por ambos clasificadores.

A continuación, y apoyándose en esta nomenclatura, se enumeran y definen matemáticamente las medidas duales recogidas en [Kuncheva & Whitaker. 2003].

- **Estadístico  $Q$** : Este estadístico, definido como:

$$Q_{ik} = \frac{N^{11}N^{00} - N^{01}N^{10}}{N^{11}N^{00} + N^{01}N^{10}} \quad [\text{Ec. 2.1}]$$

adquiere valores positivos cuando los clasificadores tienden a reconocer correctamente los mismos ejemplos y adquiere valores negativos cuando no existe correlación entre los errores cometidos por uno y otro clasificador.

- **Coefficiente de correlación,  $\rho$** : Otra forma de expresar cuantitativamente la relación entre los aciertos y los errores cometidos por dos clasificadores es mediante el coeficiente de correlación,  $\rho_{ij}$ , dado por:

$$\rho_{ij} = \frac{N^{11}N^{00} - N^{01}N^{10}}{\sqrt{(N^{11} + N^{10})(N^{01} + N^{00})(N^{11} + N^{01})(N^{10} + N^{00})}} \quad [\text{Ec. 2.2}]$$

- **Medida de desacuerdo,  $des$** : Esta medida se corresponde con el cociente entre el número de ejemplos clasificados correctamente por sólo uno de los clasificadores y el número total de ejemplos:

$$des_{ij} = \frac{N^{01} + N^{10}}{N^{11} + N^{00} + N^{01} + N^{10}} \quad [\text{Ec. 2.3}]$$

- **Medida de doble fallo,  $DF$** : Dados dos clasificadores, la medida de doble fallo se define como el cociente entre los ejemplos clasificados erróneamente por

ambos clasificadores y el número total de ejemplos que integran el conjunto de entrenamiento.

$$Dis_{ij} = \frac{N^{00}}{N^{11} + N^{00} + N^{01} + N^{10}} \quad [\text{Ec. 2.4}]$$

Adicionalmente, en [Tsymbal, et al., 2003] se recogen como medidas de diversidad duales, específicas para los problemas *multiclase*, las indicadas a continuación:

- **Estadístico kappa,  $\kappa$ :** Si  $N_{ij}$  es el número de ejemplos a los que el primer clasificador asigna la clase  $i$  y a los que el segundo clasificador asigna la clase  $j$  y  $N$  indica el total de ejemplos, entonces, se puede definir:

$$\kappa = \frac{\Theta_1 - \Theta_2}{1 - \Theta_2} \quad [\text{Ec. 2.5}]$$

donde:

$\Theta_1 = \frac{\sum_{i=1}^l N_{ii}}{N}$  es la probabilidad de que dos clasificadores coincidan en sus decisiones y

$\Theta_2 = \sum_{i=1}^l \left( \frac{N_{i*}}{N} \frac{N_{*i}}{N} \right)$  es un factor de corrección que estima la probabilidad de que los dos clasificadores coincidan en sus decisiones simplemente por azar.

El estadístico *kappa* toma valor 1 cuando los dos clasificadores coinciden en sus decisiones y toma valores negativos cuando el desacuerdo es máximo. El valor 0 refleja el caso en el que las coincidencias son fruto del azar.

- **Medida de fallo,  $f$ :** Esta medida se define como el cociente entre el número de ejemplos en los que los dos clasificadores discrepan y el número total de ejemplos. Es decir:

$$f = \frac{1}{N} \sum_{k=1}^N Dif(C_i(x_k), C_j(x_k)) \quad [\text{Ec. 2.6}]$$

donde:

$C_i(x_n)$  es la clase que el clasificador  $i$  asigna al ejemplo  $x_n$ .

Esta medida toma valor 0 cuando los clasificadores coinciden en sus predicciones y 1 cuando discrepan en todas sus decisiones.

## Medidas no duales

El objetivo de estas medidas es estimar la diversidad del conjunto considerándolo como un todo. En esta categoría se incluyen, entre otras, las siguientes medidas:

- **Ambigüedad, *amb*:** La idea que subyace a esta medida es que un problema de clasificación en el que los ejemplos pertenecen a  $K$  clases, se puede interpretar como  $K$  problemas de regresión. Por tanto, la diversidad de un conjunto

integrado por  $L$  clasificadores base se puede calcular promediando la ambigüedad de cada ejemplo sobre los distintos problemas de regresión. Así,

$$amb = \frac{1}{LNK} \sum_{l=1}^L \sum_{n=1}^N \sum_{k=1}^K \left( Is(C_l(x_n) = k) - \frac{N_k^n}{L} \right)^2 \quad [\text{Ec. 2.7}]$$

donde:

$N_k^n$ : es el número de clasificadores base que asignan clase  $k$  al ejemplo  $x_n$ .

$C_l(x_n)$ : es la clase asignada por el clasificador  $l$  al ejemplo  $x_n$ .

$Is()$ : es un predicado de verdad.

- **Entropía,  $E$ :** Para esta medida, el mayor grado de diversidad en el conjunto ocurre cuando la mitad de los clasificadores emiten una decisión y la otra mitad emiten la decisión contraria. Se trata, por tanto, de una medida enfocada, fundamentalmente, a problemas de clasificación binaria. Basándose en esta filosofía, se puede calcular la *Entropía* como:

$$E = \frac{1}{N} \sum_{k=1}^N \frac{1}{(L - [L/2])} \min\{l(x_k), L - l(x_k)\} \quad [\text{Ec. 2.8}]$$

donde:

$l(x_j)$ : Número de clasificadores que reconocen correctamente el ejemplo  $x_j$ .

$L$ : Número de clasificadores base que integran el conjunto

$N$ : Número de ejemplos de entrenamiento

Una vez vistas las técnicas que permiten obtener clasificadores base diversos y expuestas varias de las métricas que permiten cuantificar la diversidad de los mismos a continuación se analizan algunas de las heurísticas usadas para determinar qué y cuántos clasificadores base han de integrar el conjunto.

### 2.2.3. Optimización del grupo de clasificadores base

Las técnicas para obtener clasificadores diversos expuestas en el epígrafe 2.2.1. son técnicas genéricas con las que *se espera*, de manera implícita, que el grupo de clasificadores obtenidos tras su aplicación sean precisos y diversos. Sin embargo cuentan con dos limitaciones:

- Por un lado, salvo en contadas ocasiones, el número de clasificadores base que se puede encontrar es indeterminado. Así, por ejemplo, el número de clasificadores base que se puede obtener aplicando la técnica de manipulación de salidas propia del método ECOC varía entre 2 y  $2^k$  (siendo  $k$  el número de clases). Este problema se puede complicar aún más cuando, por ejemplo, la técnica usada para generar los clasificadores base es la modificación de los pesos iniciales de una red de neuronas pues, en este caso, el número de posibles clasificadores base varía entre 2 e infinito.

- Por otro lado, no hay nada que, a priori, garantice que tras obtener  $N$  clasificadores base éstos sean lo suficientemente diversos y precisos como para que al combinarlos, el conjunto de clasificadores resultante sea realmente bueno.

Para subsanar de forma conjunta estas dos limitaciones, algunos autores [Sharkey et al., 2000; Roli et al., 2001; Oliveira et al., 2003] han optado por dividir la fase de generación en dos etapas:

- 1) construcción de un grupo relativamente grande de candidatos a clasificadores base usando alguna de las técnicas ya descritas (uso de clasificadores base heterogéneos, selección de características, uso de distintos conjuntos de entrenamiento, ... ), y
- 2) selección, de entre el conjunto de posibles candidatos, del subconjunto que exhibe mayor diversidad o del subconjunto que integrado de una forma específica (apartado 2.3) deriva en una mayor precisión.

Si el grupo de candidatos a clasificadores base tiene cardinalidad  $N$ , el número de posibles subconjuntos,  $S$ , que se pueden derivar de él es:

$$S = \sum_{i=1}^N \binom{N}{i} = \sum_{i=1}^N \frac{N!}{i!(N-i)!} \quad [\text{Ec. 2.9}]$$

Esto indica que para valores de  $N$  no necesariamente muy grandes, analizar de forma exhaustiva todos y cada uno de los posibles subconjuntos es intratable. Por tanto, la obtención del mejor subconjunto ha de apoyarse en heurísticas que, en la medida de lo posible, garanticen los mejores resultados. A continuación se detallan algunas de ellas.

- *Búsqueda directa.* En este caso, el algoritmo de búsqueda comienza seleccionando un clasificador, bien de forma aleatoria o bien en función de su precisión. Una vez encontrado este clasificador, se procede a formar todos los subconjuntos de dos clasificadores que incluyen al primero. Los subconjuntos encontrados se evalúan en función de su precisión (precisión del conjunto) o de su diversidad lo que permite seleccionar aquel subconjunto que maximiza la función de evaluación establecida. Partiendo de este nuevo subconjunto, se construyen, de forma análoga, todos los subconjuntos de cardinalidad 3. Este proceso continúa hasta que los subconjuntos de tamaño  $k+1$  son peores que el mejor de los de tamaño  $k$ .

Como se ha indicado, para determinar la calidad de cualquier subconjunto de clasificadores base se debe analizar la diversidad o la precisión del conjunto de clasificadores resultante. Si se opta por determinar la bondad del subconjunto en función de su diversidad es preciso seleccionar, de entre las posibles opciones (ver epígrafe 2.2.2), la forma de cuantificar esta magnitud. Así, por ejemplo, en [Roli, et al. 2001] se usa como medida de diversidad el valor del *estadístico Q* (Ecuación 2.1.) y la *diversidad generalizada*,  $GD$  dada por:

$$GD = 1 - \frac{p_2}{p_1} \quad [\text{Ec. 2.10}]$$

donde:

$p_1$ : es la probabilidad de que un clasificador seleccionado aleatoriamente clasifique erróneamente un ejemplo seleccionado al azar

$p_2$ : es la probabilidad de que dos clasificadores seleccionados al azar clasifiquen erróneamente un ejemplo seleccionado aleatoriamente.

Por su parte, Goebel & Yan, [Goebel & Yan, 2004], usan como medida de diversidad el *coeficiente de correlación* [Ec. 2.11] dado por:

$$\rho_n = \frac{nN^f}{N - N^f - N^r + nN^f} \quad [\text{Ec. 2.11}]$$

donde:

$n$ : núm. de clasificadores que integran el subconjunto.

$N$ : núm. de ejemplos sobre los que se evalúa el conjunto.

$N^f$ : núm. de ejemplos clasificados erróneamente por todos los clasificadores.

$N^r$ : núm. de ejemplos clasificados correctamente por todos los clasificadores.

Asimismo, en [García-Muñoz, 2006] se proponen varias métricas (minimización de distancias de margen, ordenación por ángulos, ordenación basada en *Boosting*) basadas en la complementariedad existente entre un subconjunto de clasificadores y los clasificadores candidatos a formar parte del conjunto con las que se pretende fomentar el aprendizaje de los ejemplos más difíciles de reconocer. En este caso, y al igual que sucede cuando la calidad del subconjunto se mide en función de la precisión, resulta imprescindible conocer la arquitectura de integración asociada al conjunto que se desea construir (ver apartado 2.3).

- *Búsqueda inversa*: Este algoritmo parte del conjunto compuesto por todos los candidatos a clasificadores base, es decir, parte del conjunto formado por los  $N$  clasificadores base encontrados. Una vez evaluado este conjunto inicial, se construyen todos los subconjuntos que tienen un elemento menos. Si el mejor de estos nuevos subconjuntos es mejor que el conjunto inicial, se selecciona dicho subconjunto y, a partir de él, se generan todos los subconjuntos de cardinalidad  $N-2$ . El proceso finaliza cuando todos los subconjuntos de tamaño  $k$  son peores que el mejor de los de tamaño  $k+1$ .

Al igual que en el caso anterior, la calidad de cualquier subconjunto se determina en función de su precisión o de su diversidad [Roli et al., 2001], [Hao et al., 2003].

- *Búsqueda basada en Algoritmos Genéticos*. Los Algoritmos Genéticos (AG's) [Goldberg, 1989; Srinivas & Patnaik, 1994] son procesos de búsqueda basados en la teoría de la evolución de Darwin, apropiados para resolver problemas donde el espacio de soluciones puede resultar demasiado extenso. A partir de una población inicial de soluciones tentativas a un problema, se crean sucesivas



generaciones seleccionando, combinando (cruzando) o modificando (mutando) soluciones de la generación anterior. El proceso finaliza cuando se obtiene una solución que satisface ciertos criterios o tras un número determinado de generaciones. En el caso que aquí se presenta (selección del mejor subconjunto de clasificadores base), los individuos que componen la población están representados por cadenas de bits en las que, de una u otra forma, se codifica la presencia o ausencia de los distintos candidatos a clasificadores base y la bondad de cada posible solución (*fitness*) se mide en función de la diversidad o precisión del conjunto [Giacinto, 1998; Hao et al., 2003; Zhou & Tang, 2003; Ledezma, 2004; Ordóñez et al., 2008].

Oliveira, [Olivera et al, 2003], ve el problema de encontrar el mejor subconjunto de clasificadores como un problema *multiobjetivo* que se puede resolver usando un Algoritmo Genético que integre el concepto de *Optimización de Pareto*. Así, una vez encontrado el conjunto de candidatos a clasificadores base, el siguiente paso es aplicar un Algoritmo Genético Multiobjetivo que intenta encontrar el subconjunto de clasificadores base más diverso y preciso.

- *Métodos basados en aprendizaje no supervisado*. El método de selección propuesto en [Roli et al., 2001] se sustenta en la idea de que cualquier conjunto compuesto por un número elevado de clasificadores base puede, en función del número de errores coincidentes, descomponerse en  $M$  subconjuntos disjuntos,  $C_i$ . Bajo esta hipótesis, para encontrar un grupo de clasificadores base diversos, bastaría con identificar los clasificadores base pertenecientes a cada subconjunto  $C_i$  y extraer, de cada uno de ellos, un clasificador. El conjunto,  $C^*$ , así formado estará, por tanto, compuesto de clasificadores base diversos por lo que es esperable que al integrarlos den lugar a un conjunto de clasificadores preciso. Otros métodos basados en aprendizaje no supervisado en los que el agrupamiento se realiza a partir de la salida dada por cada clasificador sobre cada ejemplo, son los propuestos en [Lazarevic & Obradovic, 2001] y en [Fu, et al., 2005].

### 2.3. Fase de integración de los clasificadores base

La construcción de un conjunto de clasificadores implica generar una serie de clasificadores base diversos y precisos y, además, seleccionar un método de integración que combine las hipótesis asociadas a cada clasificador base y genere una hipótesis más precisa que cualquiera de las hipótesis individuales.

En función de su arquitectura, los esquemas usados en la fase de integración se pueden clasificar en tres grandes categorías [Jain et al., 2000]: *paralela*, *cascada* o *serie*, y *jerárquica*. A continuación se describe cada una de ellas.

#### Arquitectura paralela

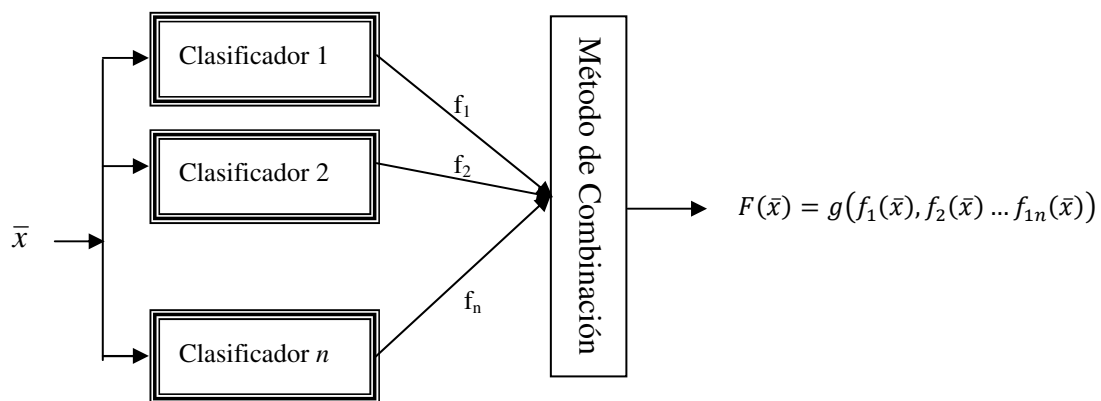
En la *arquitectura paralela* los clasificadores base son independientes unos de otros por lo que, ante un patrón de entrada todos emiten una decisión. Según la política usada para combinar estas decisiones y conseguir una única decisión final, las estrategias de

combinación propuestas en la literatura se pueden agrupar en técnicas de *fusión* y técnicas de *selección*.

La *fusión* asume que todos los clasificadores base intervienen en la decisión final del sistema. Se trata por tanto de una estrategia de combinación cooperativa y competitiva. Es cooperativa en el sentido de que todas las decisiones intervienen en la decisión final del sistema. Es competitiva debido a que, en la decisión final, unas decisiones prevalecen sobre otras. Los conjuntos de clasificadores que se ajustan a este esquema de integración requieren, por tanto, de un módulo de combinación capaz de integrar las decisiones dadas por todos los clasificadores base y emitir una decisión final. Este módulo puede ser una simple función matemática o un *meta-clasificador*, es decir, un clasificador entrenado para combinar las salidas de los distintos clasificadores base.

Cuando la decisión final del sistema es el resultado de combinar matemáticamente las salidas de los componentes individuales, los principios más comúnmente usados son la votación por mayoría simple y la votación por mayoría ponderada. En la votación por mayoría simple, la clasificación del sistema coincide con la dada por un mayor número de clasificadores base, recurriéndose a la selección aleatoria en caso de empate. La votación por mayoría ponderada es una variante de la votación simple en la que la cada decisión está afectada por un peso diferente. Aunque, como se ha indicado, la mayoría simple o ponderada es la técnica de integración más usada, algunos sistemas determinan la decisión final del sistema por promedio, minoría, mediana o producto de votos.

En la Figura 2.1 se muestra un esquema de la integración en paralelo de los clasificadores base.



**Figura 2.1.** Conjunto de clasificadores integrados en paralelo. En este caso, la salida del sistema se determina combinando las salidas generadas por los clasificadores base.

El uso de funciones matemáticas tiene como punto débil el que si la mayoría de los clasificadores realizan predicciones erróneas, la clasificación final del sistema será, casi con total seguridad, errónea. Para tratar este problema, en los últimos años, se ha popularizado el uso de los denominados meta-clasificadores, es decir, clasificadores que intentan modelizar cómo han de combinarse las decisiones de los clasificadores base.

*Stacking* [Wolpert, 1992] es, quizás, el método más conocido que introduce el concepto de meta-clasificador.

Las técnicas de *selección* asumen que cada clasificador base es un *experto* en una determinada región del espacio [Zhu et al., 2004]. Por tanto, a la hora de clasificar una nueva instancia solo se considera la decisión emitida por un único clasificador. Dependiendo de si la región de competencia de un clasificador se define durante la fase de entrenamiento o durante la fase de clasificación, las técnicas de selección se dividen en estáticas o dinámicas [Kuncheva, 2002].

Cuando se usa selección estática, antes de clasificar un nuevo ejemplo es necesario determinar la región en la que el ejemplo está incluido y el clasificador responsable de dicha región. Por tanto, el punto crítico de estas técnicas es establecer el conjunto de regiones y determinar el clasificador asociado a cada región. En [Kuncheva, 2002] se propone un método en el que las regiones se determinan aplicando el algoritmo *k-medias* al conjunto de datos de entrenamiento y la asignación región-clasificador se establece localizando el clasificador más preciso en cada una de las regiones.

En la selección dinámica, tal y como ya se ha señalado, la elección del clasificador base que debe etiquetar a un ejemplo dado se realiza durante la fase de clasificación. Normalmente, esta elección está basada en el grado de certeza de los distintos clasificadores base, es decir, el clasificador encargado de emitir la decisión será aquel que, en la región de influencia, se considera más seguro. En [Woods et al., 1997] se propone considerar la región de influencia como el subconjunto formado por los *k* ejemplos de entrenamiento más próximos al que se desea clasificar y determinar el grado de certeza de los distintos clasificadores base en función de su precisión sobre los ejemplos de entrenamiento contenidos en dicha región de influencia. Para determinar la precisión, los mencionados autores proponen dos alternativas: considerar todos los ejemplos contenidos en la región de influencia o considerar solo aquellos ejemplos cuya clase coincide con la asignada al ejemplo que se desea clasificar.

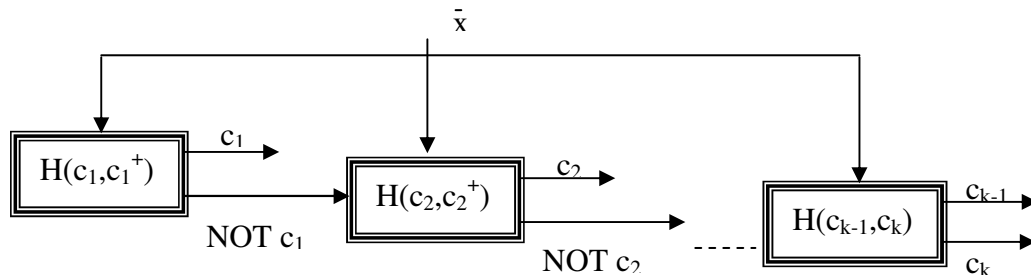
### Arquitectura en cascada

En la *arquitectura en cascada* los clasificadores base se invocan secuencialmente hasta que el patrón de entrada se considera clasificado. Por tanto, si la decisión emitida por uno de los clasificadores se considera fiable, el proceso de clasificación se detiene y la salida del conjunto coincide con la emitida por este clasificador. En caso contrario, es decir, si tras la llamada a un clasificador éste no puede determinar la clase a la que pertenece el patrón de entrada o su decisión no goza del suficiente crédito, dicho patrón será enviado al siguiente clasificador para que éste emita su decisión. Se trata, por tanto, de una estrategia competitiva en el sentido de que la salida del conjunto coincide con la salida dada por un único clasificador base. En concreto, por el invocado en último lugar.

Un ejemplo representativo de esta arquitectura es la metodología OAHO (*One-Against-Higher-Order*) [Ou & Murphey, 2007; Murphey et al., 2007] usada para resolver problemas multiclase cuando los datos de entrenamiento no están balanceados. La aplicación de esta metodología pasa por ordenar las clases en función del número de ejemplos de entrenamiento que las representan, i.e.,  $C=\{c_1, c_2, \dots, c_k\}$  tal que  $|\Omega_i| \geq |\Omega_{i+1}|$

siendo  $\Omega_i$  el conjunto de datos de entrenamiento asociados a la clase  $c_i$  y  $|\Omega_i|$  la cardinalidad de dicho conjunto. Una vez ordenadas las clases, se construyen los clasificadores base de forma que el clasificador  $i$ -ésimo,  $H(c_i, c_i^+)$ , se especializa en distinguir los ejemplos asociados a la clase  $c_i$  de los asociados a las clases de orden superior, es decir, se entrena considerando que los ejemplos asociados a la clase  $c_i$  son positivos y los asociados a las clases de orden superior,  $c_i^+$ , son ejemplos negativos. Una vez contruidos los clasificadores, éstos se integran de forma que el clasificador  $i$ -ésimo,  $H(c_i, c_i^+)$ , solo emite su decisión respecto al ejemplo  $\bar{x}$  si y solo si los clasificadores  $H(c_j, c_j^+)$  con  $j \geq i$ , han clasificado el ejemplo como negativo.

En la Figura 2.2 se muestra un esquema representativo de la arquitectura en cascada.



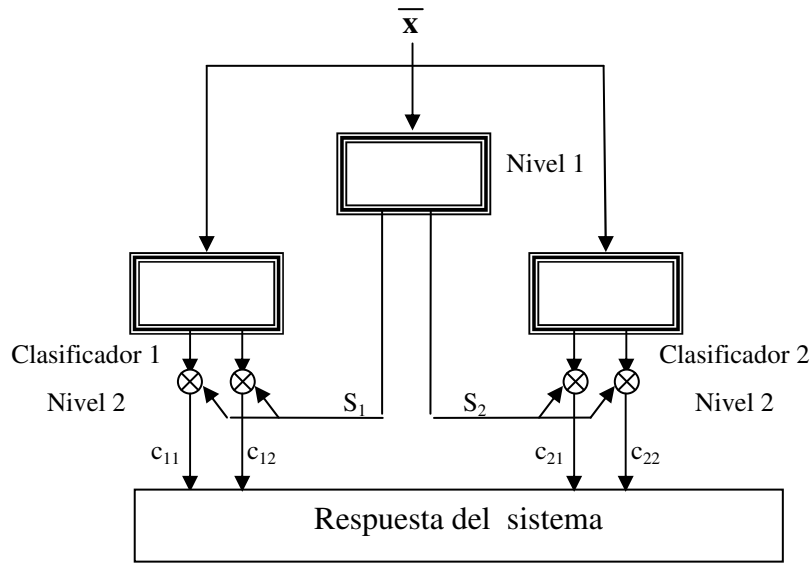
**Figura 2.2.** Conjunto de clasificadores integrados en serie que resuelve un problema multiclase aplicando la metodología OAHO.

### Arquitectura jerárquica

En la *arquitectura jerárquica* los clasificadores se organizan en una estructura con forma de árbol, en la que la llamada a un clasificador base depende de la salida dada por los clasificadores que, jerárquicamente, le preceden. La idea que subyace a esta arquitectura es que una tarea de clasificación relativamente compleja puede ser reemplazada por una combinación de clasificaciones más simples en la que la viabilidad de una clasificación está supeditada a los resultados de las clasificaciones anteriores.

Una aplicación representativa de la combinación jerárquica (Figura 2.3) es la propuesta en [Poddar & Rao, 1993] para la resolución de un problema de clasificación en el que los ejemplos pertenecen a  $k$  clases distintas. Para resolver este problema, los autores agrupan las clases en  $S$  conjuntos disjuntos de cardinalidad  $s_i$ ,  $\sum_{i=1}^S s_i = k$ , y aplican una

estructura jerárquica de 2 niveles. El primer nivel consta de un clasificador, entrenado para determinar el conjunto al que pertenece la clase que se asignará al ejemplo que se quiere clasificar. El segundo nivel está compuesto de  $S$  clasificadores, entrenados para determinar cuál de las clases que integran el conjunto identificado en el primer nivel es a la que, específicamente, corresponde el ejemplo dado. Se trata, por tanto, de un esquema en el que la llamada a un clasificador del segundo nivel depende del resultado dado por el clasificador del primer nivel, es decir, de un esquema de integración jerárquico. Además, los clasificadores son incompatibles entre sí en el sentido de que cada uno de ellos se entrena con un conjunto de datos de entrenamiento distinto.



**Figura 2.3.** Conjunto de clasificadores integrados jerárquicamente usado en la resolución de un problema de 4 clases. El primer nivel está formado por un clasificador de 2 salidas y el segundo nivel por dos clasificadores con 2 salidas cada uno. La respuesta del sistema se obtiene multiplicando las salidas de ambos niveles. En este esquema el conjunto de clases se supone dividido en dos subconjuntos ( $S_1=\{c_{11},c_{12}\}$ ,  $S_2=\{c_{21},c_{22}\}$ ) cada uno de los cuales contiene 2 clases.

## 2.4. Estudios comparativos

A pesar del gran interés que en las últimas décadas han suscitado los conjuntos de clasificadores y los muchos trabajos realizados sobre el tema [Kuncheva, 2005; Sansone, et al., 2011; Zhang & Yunquian, 2012], existen pocos estudios enfocados a la comparación de los distintos métodos. La justificación estriba en que la gran variedad de técnicas aplicables, tanto en la fase de selección como en la de integración, da como resultado un muy amplio abanico de arquitecturas que impide toda comparación exhaustiva. A esto hay que sumar que al igual que sucede con los sistemas de clasificación clásicos (redes de neuronas, árboles de decisión, modelos bayesianos, ...) la calidad de un conjunto de clasificadores depende en gran medida del problema al que se aplique y, por tanto, intentar obtener conclusiones generales es prácticamente una utopía.

Uno de los estudios, a nuestro juicio, más interesante y completo, es el expuesto en [Duin & Tax, 2000]. En este trabajo, se comparan los resultados obtenidos al aplicar distintas técnicas de fusión a varios conjuntos de clasificadores estructurados en paralelo. En concreto, este estudio analiza la salida del conjunto cuando ésta se obtiene por aplicación de funciones matemáticas -mayoría, máximo, mediana, media, mínimo o producto- y cuando la función se induce aplicando determinados algoritmos de aprendizaje -modelo Bayesiano, regla del vecino más cercano o mínima distancia a la media (*Nearest Mean*)-. Además, contempla: conjuntos formados por clasificadores del mismo tipo entrenados sobre ejemplos descritos por distintos subconjuntos de

características, conjuntos formados por clasificadores heterogéneos entrenados con ejemplos descritos por un mismo conjunto de características y conjuntos en los que los clasificadores base son, a su vez, conjuntos de clasificadores.

El conjunto de datos utilizado en la realización de este estudio se denomina “*Multiple Features*”, se encuentra disponible en el repositorio de datos del UCI [Frank & Asuncion, 2010] y consta de 2000 ejemplos, cada uno de los cuales representa un dígito (‘0’ a ‘9’) manuscrito.

Los resultados obtenidos demuestran que, sobre esta base de datos, los conjuntos compuestos por clasificadores homogéneos entrenados con ejemplos descritos por distintos subconjuntos de características, son más precisos que los conjuntos heterogéneos contruidos con ejemplos descritos por el mismo conjunto de características. Es más, muchos de los conjuntos heterogéneos son menos precisos que el mejor de los clasificadores base que lo integran. Por otro lado, concluyen que no existe una supremacía general de ninguno de los métodos de combinación considerados si bien los *metaclasificadores* (clasificadores de segundo nivel) basados en el vecino más cercano y la mínima distancia a la media parecen ser bastante estables. Finalmente, indican que la arquitectura más precisa es aquella en la que los clasificadores base son, por sí mismos, conjuntos de clasificadores.

Otro estudio relevante es el presentado en [Sharkey et al., 2000] en el que, de forma exhaustiva, se comparan 300 conjuntos de clasificadores formados por 3 redes de neuronas e integrados por voto mayoritario. La diversidad en los clasificadores base se obtiene modificando la topología de la red, el número de características con las que se describen los ejemplos y/o el subconjunto de datos de entrenamiento usado en la construcción del clasificador. Tras analizar los resultados obtenidos, los autores concluyen que, en media, los mejores conjuntos derivan de combinar clasificadores base contruidos a partir de ejemplos descritos por distintos conjuntos de características.

En una línea bastante distinta están los estudios cuya finalidad es establecer una relación entre la diversidad de los clasificadores base y la precisión del conjunto. De entre ellos, el más representativo es el expuesto en [Kuncheva & Whitaker, 2003] en el que se comparan 10 medidas de diversidad y se intenta establecer una relación entre ellas y la precisión del conjunto. El estudio concluye admitiendo que la relación entre las distintas medidas de diversidad y la precisión del conjunto es un tema abierto. Es decir, no existe una medida de diversidad con la que, en función de un valor concreto, se garantice una buena precisión del conjunto.

Otros estudios presentes en la Bibliografía, analizan las ventajas y desventajas que, a la hora de resolver un problema *multiclase* presentan los distintos métodos de descomposición binaria. Así, en [García-Pedrajas & Haro, 2007] se analiza, sobre 41 bases de datos distintas, el comportamiento de OAO, OAA y ECOC, cuando el algoritmo usado para la construcción de los distintos clasificadores binarios es C4.5. En este trabajo se concluye que, sobre la mayor parte de los problemas analizados, ECOC y OAO presentan mayor poder de generalización que OAA y C4.5 (en su versión *multiclase*). No obstante, al incorporar en el estudio otros algoritmos de aprendizaje observan que la calidad de los distintos métodos de descomposición puede estar condicionada por el algoritmo usado en la construcción de los clasificadores base. Por su parte, en [Ou & Murphey, 2007] se muestra un estudio en el que se analiza el

comportamiento de varios sistemas basados en redes de neuronas artificiales sobre distintos problemas *multiclase*. En concreto, en este trabajo, se analiza, a distintos niveles, el comportamiento de los sistemas basados en descomposición binaria (OAA, OAO, OAH0 y ECOC) y el de una red de neuronas simple cuando ésta se construye a partir de ejemplos en los que la clase se codifica usando la metodología OAA o la metodología ECOC. Al analizar el comportamiento de los distintos sistemas sobre bases de datos con características dispares (alto desbalanceo entre clases, número reducido de ejemplos de entrenamiento, elevado número de clases, ...) concluyen que la calidad de cada modelo está fuertemente condicionada por las peculiaridades del problema sobre el que se aplica. Así, por ejemplo, la idoneidad de los modelos compuestos por una única red de neuronas parece estar limitada a problemas en los que tanto el número de ejemplos como de clases es pequeño. Por el contrario, cuando el número de clases es alto, la arquitectura que parece presentar una menor tasa de error es OAO pero también es la arquitectura que suele requerir un mayor tiempo de aprendizaje. Por último, y siempre según este estudio, en problemas en los que existe desbalanceo entre clases la arquitectura más recomendable es OAH0. No obstante, en ciertos problemas, la tasa de error de esta arquitectura es superior a la lograda con una red de neuronas simple construida con ejemplos sobre los que se ha eliminado el desbalanceo replicando datos de las clases minoritarias.

A la vista de estos y otros estudios se puede concluir que la construcción de un conjunto de clasificadores realmente óptimo en todas las situaciones es una tarea que, por ahora, está lejos de ser resuelta, lo que justifica el trabajo que se propone en este trabajo de Tesis Doctoral.





## Capítulo 3

### Objetivos de la Tesis Doctoral

El objetivo general que persigue esta Tesis Doctoral es definir nuevas arquitecturas de conjuntos de clasificadores que permitan resolver problemas de clasificación supervisada en los que las instancias están descritas por un elevado número de características y en los que el criterio de clasificación engloba a más de dos clases.

Según lo expuesto en los capítulos precedentes, para lograr que un conjunto de clasificadores sea más preciso que cualquiera de sus miembros, es necesario que estos últimos sean precisos y diversos. No obstante, y dado que a nivel de clasificador base es difícil encontrar clasificadores muy precisos que, a su vez, sean muy diversos entre sí, se impone como objetivo que las arquitecturas propuestas sean complementarias en este sentido. Así, en este trabajo se presentan **dos** nuevas arquitecturas de conjuntos de clasificadores en una de las cuales, denominada BCE -*Binary Complementary Ensemble*-, sin obviar la diversidad, se fomenta la precisión de los clasificadores base, mientras que en la otra, denominada CCE -*Complementary Complementary Ensemble*-, se fomenta la diversidad frente a la precisión. Las diferencias y la complementariedad existente entre ambas arquitecturas permitirá analizar la influencia que, en el comportamiento global del conjunto, tiene la primacía de la precisión frente a la diversidad y viceversa, y determinar si, bajo determinadas circunstancias, la aplicación de una arquitectura puede ser más acertada que la otra. Por tanto, uno de los objetivos específicos de este trabajo es analizar y comparar la calidad de las propuestas realizadas cuando éstas se enfrentan a problemas caracterizados por:

- a) Un número reducido de clases.
- b) Un número no excesivamente alto de atributos.
- c) Una distribución de clases no homogénea.
- d) Un gran número de ejemplos de entrenamiento.
- e) Una estructura de clases jerárquica.

Además, a partir de las arquitecturas propuestas se pretende construir un sistema de reconocimiento de señales de tráfico que pueda ser implementado como una solución *hardware* y formar parte de un *Sistema Avanzado de Asistencia a la Conducción* (ADAS). Para satisfacer este requisito, se hace imprescindible que las arquitecturas propuestas estén basadas en *redes de neuronas artificiales* de dimensiones reducidas. La aplicación de los modelos propuestos a la resolución de este problema no sólo permitirá determinar si los mismos constituyen una alternativa válida para la construcción de un sistema implementable en *hardware* sino que, también, permitirá analizar la robustez de ambos cuando las instancias son imágenes que sufren de oclusiones, sombras y/o falta de nitidez.

Hasta la fecha, la mayor parte de los conjuntos de clasificadores recogidos en la Bibliografía combinan clasificadores base en los que la diversidad se consigue utilizando una de las técnicas de generación de diversidad recogidas en el Capítulo 2 de este trabajo. Dado que la diversidad es una condición necesaria (aunque no suficiente) para conseguir un conjunto de clasificadores óptimo, parece oportuno diseñar nuevas arquitecturas en las que la diversidad de los clasificadores base esté inducida, de forma simultánea, por distintas técnicas. Por tanto, uno de los objetivos específicos que se plantean en este trabajo es la obtención de clasificadores base diversos fusionando varias de las técnicas que, a tal efecto, se recogen en la Bibliografía. En concreto, y siguiendo la terminología recogida en [Dietterich, 1997], los conjuntos de clasificadores que se proponen en este trabajo logran la diversidad:

- Modificando las etiquetas que describen la clase asociada a los distintos ejemplos [Ou & Muprhey, 2007].
- Variando el conjunto de características con las que se describen los ejemplos [Optiz, 1999; Bryll, et al., 2003].
- Modificando el subconjunto de ejemplos usado en el entrenamiento [Breiman, 1996; Parmanto et al., 1996; Schapire, 1990].
- Incorporando aleatoriedad en el algoritmo de aprendizaje [Kolen & Pollack, 1990].

Por otro lado, muchos de los conjuntos de clasificadores descritos en la Bibliografía, han sido explícitamente diseñados para resolver problemas binarios. Por este motivo, su aplicación a problemas *multiclase* exige introducir ciertas modificaciones que en ocasiones implican un coste computacional demasiado elevado, o a fusionar estos modelos con técnicas de descomposición binaria. Por ello, otro de los objetivos que se abordan en este trabajo es el diseño de modelos concebidos explícitamente para la resolución de problemas *multiclase* en los que tanto la complejidad como el coste computacional sean moderados y, por tanto, permitan la construcción de clasificadores capaces de trabajar en tiempo real.

Recapitulando, y a modo de síntesis, se puede indicar que el objetivo principal de esta Tesis es definir dos nuevas arquitecturas de conjuntos de clasificadores:

- Diseñados para la resolución de problemas *multiclase* en los que los ejemplos están descritos por un elevado número de atributos.
- Constituidos por redes de neuronas artificiales.
- Con complejidad y coste computacional moderados.

- La diversidad se induce fusionando varias de las técnicas recogidas en la Bibliografía.
- A nivel de clasificador base, se prima la diversidad frente a la precisión y viceversa.

Además, y de modo empírico, se pretende comprobar si ambas arquitecturas se pueden considerar *de propósito general* o si, por el contrario, bajo determinadas circunstancias la aplicación de una de ellas resulta más adecuada que la otra.



## Capítulo 4

### Descripción de los Modelos Propuestos: BCE y CCE

En los capítulos precedentes se ha definido el concepto de conjunto de clasificadores y se han descrito algunas de las técnicas de generación de conjuntos más usadas en las últimas décadas. Algunos de los modelos presentados, OAO (del inglés *One Against One* –uno contra uno–), OAA (del inglés *One Against All* –uno contra todos–) o ECOC (*Error Correcting Output Code*), han sido explícitamente diseñados para resolver problemas *multiclase* -los ejemplos pertenecen exactamente a una de  $k$  clases ( $k \geq 2$ )- y basan su metodología en la descomposición del problema en subproblemas dicotómicos. Otros, como *Boosting* fueron originalmente diseñadas para resolver problemas binarios. Por ello, su aplicación a problemas *multiclase* ha obligado a introducir ciertas modificaciones que en ocasiones implican un coste computacional demasiado elevado [Freund & Schapire, 1997] o a fusionar ADABOOST (versión más extendida de *Boosting*) con algunas de las técnicas de descomposición binaria previamente mencionadas [Shapire, 1997; Allwein et al, 2000]. Otras sistemas como *Bagging* o *Stacking* se consideran técnicas genéricas que, dependiendo de las características de los clasificadores base, pueden aplicarse tanto a dominios binarios como a dominios *multiclase*. No obstante, dado que la construcción de un clasificador *multiclase* suele resultar mucho más compleja y computacionalmente más costosa que la construcción de un clasificador binario, un gran número de las investigaciones enmarcadas en el aprendizaje *multiclase* se han centrado en el uso de modelos basados en descomposición binaria [Tax & Duin, 2002; Kraipeerapun et al, 2006; Ou & Murphey, 2007; Lézoray & Cardot, 2008; García-Pedrajas & Ortiz-Boyer, 2011; Galar et al, 2011; Oong & Mat Isa, 2012].

En este capítulo se presentan dos nuevos modelos de conjuntos de clasificadores, BCE (*Binary-Complementary Ensemble*) y CCE (*Complementary-Complementary Ensemble*), diseñados fundamentalmente para resolver problemas de clasificación en los que el número de clases es finito y superior a dos y los ejemplos están descritos por un gran número de características. Ambos modelos intentan superar las limitaciones que

presentan los sistemas basados en descomposición binaria y más concretamente los referidos a la arquitectura OAA que se detallan en el siguiente epígrafe.

#### 4.1. Introducción

A la hora de construir un clasificador *multiclase*, el esquema más simple, más usado y probablemente también el más criticado, es el denominado *uno contra todos* (*one against all*, OAA) [Rifkin & Klautau, 2004; Ou & Murfhey, 2007]. En este modelo, el problema de  $k$  clases se descompone en  $k$  problemas de clasificación binarios, cada uno de los cuales distingue, o separa, una clase de las  $(k-1)$  clases restantes. Los clasificadores binarios así contruidos son capaces de indicar si un ejemplo pertenece o no a su clase asociada pero, en el supuesto de que el mencionado ejemplo sea clasificado como perteneciente a cualquiera de las  $k-1$  clases restantes, no podrá indicar a qué clase específica pertenece. Esta característica, implícita a la arquitectura *uno contra todos*, hace que los errores cometidos por un único clasificador sean difíciles de subsanar. Es decir, la diversidad de los clasificadores base que integran el conjunto no siempre conlleva una mejora en la clasificación de este último. A continuación se analiza este aspecto en más detalle:

Supongamos una arquitectura *uno contra todos* integrada por  $k$  clasificadores dicotómicos con salida binaria ( $y_i \in \{0, 1\}$ ,  $i \in \{1, 2, \dots, k\}$ ) y consideremos que, al intentar clasificar un determinado ejemplo, solo uno de estos clasificadores lo cataloga de forma errónea, es decir, los clasificadores base son diversos. Bajo estas condiciones, las posibles situaciones que se pueden presentar son:

1. El clasificador,  $C_j$ , que comete el error, clasifica como negativo ( $y_j=0$ ) un ejemplo que, para él, debiera ser positivo (*falso negativo*). En este caso, y puesto que para el resto de clasificadores dicho ejemplo es negativo, la salida de todos ellos será  $\{0\}$  y, por tanto, la salida global del conjunto será  $\{0, \dots, 0, \dots, 0\}$ . Es decir, a menos que la clase a la que pertenece el ejemplo se determine de forma aleatoria, no hay mecanismos para determinar la clase a la que pertenece el ejemplo y, por tanto, el error no es subsanable.
2. El clasificador,  $C_j$ , clasifica como positivo un ejemplo que, para él, debiera ser negativo (*falso positivo*). En este caso, y admitiendo que el ejemplo a clasificar pertenece a la clase  $c_i$  ( $i \neq j$ ), existirán dos clasificadores,  $C_i$  y  $C_j$ , que clasificarán el ejemplo como perteneciente a su clase, mientras que el resto de clasificadores lo catalogarán como negativo. Es decir, la salida global del conjunto será  $\{0, \dots, 1, \dots, 1, \dots, 0\}$ . Por tanto, y al igual que en el caso anterior, será imposible determinar de forma inequívoca a qué clase pertenece el ejemplo.

De este análisis se puede concluir que la fiabilidad de la arquitectura *uno contra todos* depende fundamentalmente de la precisión de los clasificadores dicotómicos que lo integran pero no de su diversidad. Es decir, bajo los supuestos anteriores, subsanar los errores cometidos por cada uno de los clasificadores binarios depende del azar por lo que, aunque los clasificadores dicotómicos sean diversos, el error del sistema puede llegar a coincidir con la suma de los errores individuales.

Para tratar este problema, un gran número de esquemas basados en esta arquitectura admiten que la salida de los clasificadores binarios,  $y_i$ , sea un valor incluido en el rango  $[0,1]$  y que la clase asociada a un ejemplo dado,  $\bar{X}$ , se corresponda con la clase asociada al clasificador que genera la salida de mayor valor [Galar et al., 2011]. Es decir:

$$C(\bar{x}) = F(\bar{x}, y_1, y_2, \dots, y_k) = \arg \max_{i=1, \dots, k} (y_i) \quad [\text{Ec. 4.1}]$$

Esta forma de tratar las salidas evita situaciones de empate y crea cierta dependencia entre la diversidad de los clasificadores y la precisión del conjunto. Sin embargo, la correcta clasificación de un ejemplo depende fundamentalmente del valor de salida dado por el clasificador asociado a la clase a la que pertenece el ejemplo. Es decir,  $\bar{X} \in c_i$  será clasificado correctamente por el conjunto si y solo si la salida,  $y_i$ , dada por el clasificador asociado a la clase  $c_i$ , es mayor que cualquiera de las salidas,  $y_k$ , dadas por el resto de clasificadores. En consecuencia, la dependencia entre la diversidad de los clasificadores base y la precisión del conjunto no siempre está garantizada pues depende de un criterio ( $y_i > y_j$ ) que no suele estar implícito en los algoritmos de aprendizaje.

En este trabajo, y como primera propuesta se detalla una modificación de la arquitectura *uno contra todos* con la que se pretende garantizar que la diversidad de los clasificadores base repercute en un aumento en la precisión del conjunto. Para alcanzar este objetivo, los clasificadores base del conjunto estarán formados por la unión de un clasificador binario -análogo a los usados por OAA- y un clasificador *multiclase*.

Otro de los aspectos que paralelamente se plantea en este trabajo, es la necesidad de construir un sistema capaz de resolver problemas en los que los ejemplos están descritos por un elevado número de características. En contra de lo que podría parecer, cuando el número de características que definen los ejemplos es elevado, una reducción del conjunto de características no siempre conlleva una pérdida en la precisión del sistema sino que, en ocasiones, puede dar lugar a un sistema más preciso [Liu & You, 2002; Oliveira et al., 2003; Kim et al., 2006]. Esta mejora en la precisión viene determinada por la redundancia o la irrelevancia de determinadas características. En otras palabras, cuando el número de características que describen los ejemplos es elevado, la probabilidad de que entre las mismas exista un alto porcentaje de características irrelevantes o redundantes es bastante elevada. En estos casos, el aprendizaje suele convertirse en una tarea más costosa y, en ocasiones, menos precisa. Además, la elección de un reducido subconjunto de características puede convertirse en un requisito imprescindible en determinados dominios. Así, por ejemplo, cuando el sistema de clasificación diseñado ha de implementarse como una solución hardware (p.e., implementación de una red de neuronas artificial sobre FPGA -*Field Programmable Gate Array*-), un elevado número de características hace inviable esta implementación [Muthuramalingam, et al., 2007].

Una vez detallados los principales problemas de la arquitectura OAA y expuesta la necesidad de incluir un proceso de selección de características, a continuación se presenta la primera de las propuestas de esta Tesis Doctoral.

## 4.2. Descripción de BCE

El sistema BCE (*Binary-Complementary Ensemble*) se construye sobre la arquitectura *uno contra todos* y pretende garantizar que la diversidad de los clasificadores base repercute en un aumento de la precisión del conjunto.

Atendiendo a la terminología recogida en [Dietterich, 1997], BCE es un conjunto de clasificadores en el que la diversidad se consigue:

- i. Modificando las etiquetas que describen la clase asociada a los distintos ejemplos [Ou & Murphey, 2007].
- ii. Variando el conjunto de características con las que se describen los ejemplos [Optiz, 1999; Bryll, et al., 2003].
- iii. Modificando el subconjunto de ejemplos usado en el entrenamiento [Breiman, 1996; Parmanto et al., 1996; Schapire, 1990].
- iv. Incorporando aleatoriedad en el algoritmo de aprendizaje [Kolen & Pollack, 1990].

A diferencia de los métodos *multiclase* basados en descomposición binaria (OAA, OAO o ECOC [Dietterich & Bakiri, 1995]; [Ou & Murphey, 2007]), el conjunto de clasificadores que proponemos, pasa por admitir que los clasificadores que integran el conjunto no se conforman con indicar si un ejemplo pertenece o no a una o varias clases específicas sino que intentarán indicar explícitamente la clase concreta a la que pertenece la instancia.

Al igual que en la arquitectura *uno contra todos*, el número de clasificadores que integrarán el conjunto coincide con el número de clases que describen los ejemplos. La principal diferencia respecto a la arquitectura *uno contra todos* es que, en este nuevo modelo, los clasificadores base no serán clasificadores binarios, sino la unión de un clasificador binario y un clasificador *multiclase*. Para simplificar la terminología, los clasificadores *multiclase* serán denominados *clasificadores complementarios* y los módulos compuestos por un clasificador binario y su correspondiente clasificador complementario se denominarán *módulos base*.

Por otro lado, y puesto que una de las motivaciones de este trabajo es desarrollar un sistema que, una vez diseñado, pueda ser implementado en *hardware*, previo a la construcción de cada clasificador se procederá a seleccionar el conjunto de características que, en cada caso, mejor definen los ejemplos de aprendizaje. A diferencia de otros modelos, el conjunto de características que define un ejemplo no será único, sino que, dependerá del clasificador al que, en un momento dado, se vincule el ejemplo. Desde un punto de vista *software*, esta reducción en el número de características permitirá construir clasificadores más simples y, por tanto, sistemas en los que tanto el tiempo de aprendizaje como el de clasificación se verá reducido considerablemente.

### 4.2.1. Construcción de los módulos base

Los módulos base que integran BCE (Figura 4.1) están compuestos por dos clasificadores. El primero de ellos,  $B_i$ , será un clasificador binario entrenado para



distinguir si un ejemplo pertenece o no a una determinada clase. El segundo clasificador, denominado *clasificador complementario* y representado por  $\bar{C}_i$ , será un clasificador *multiclase* con  $k-1$  salidas. Como se indica más adelante (sec. 4.2.2), este clasificador sólo intervendrá bajo determinadas circunstancias y su actuación permitirá decidir a cuál de las  $k-1$  clases restantes pertenece el ejemplo que se desea clasificar.

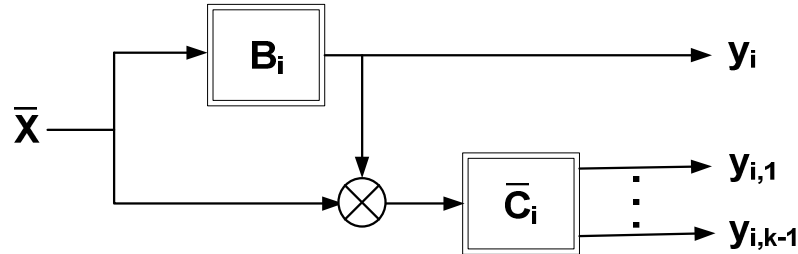


Figura 4.1. Módulo base asociado a la  $i$ -ésima clase.

A continuación se detalla cómo se han construido los clasificadores que integran cada uno de los módulos base.

### Clasificadores Binarios

Como se ha señalado, los clasificadores binarios que integran el conjunto, son análogos a los usados en la arquitectura *uno contra todos*. Por tanto, todos ellos se construirán a partir del mismo conjunto de ejemplos pero, en cada caso, con una codificación de clases distinta. En concreto, la clase asociada a los ejemplos usados en la construcción del clasificador binario vinculado a la  $i$ -ésima clase se representará con único dígito cuyo valor será  $1$  si el ejemplo pertenece a esta clase y  $0$  en caso contrario.

Una de las premisas que ha de satisfacer este modelo es la capacidad para trabajar con ejemplos descritos por un gran número de características. Aprender de este tipo de ejemplos suele ser una tarea costosa y, en ocasiones, la redundancia o irrelevancia asociada a las características con las que se describen los ejemplos hace que el aprendizaje sea poco preciso. Por tanto, para facilitar y mejorar el proceso de aprendizaje, una vez recodificada la clase asociada a cada ejemplo, se procederá a determinar el subconjunto de características más relevantes. En este trabajo, y a menos que se indique lo contrario, el proceso de selección de características se ha realizado usando la combinación *Correlation-based Feature Selection* (CFS) [Hall, 1999] y *Best First* (BF) [Russell & Norvig, 2003] implementada en la herramienta *Weka* [Witten & Frank, 2005]. CFS es un método de evaluación de atributos que premia aquellos atributos que tienen una alta correlación con la clase pero poca correlación entre ellos. Por su parte, BF es un método de búsqueda que analiza lo que mejora o empeora un grupo de atributos al añadir nuevos elementos y que permite hacer retrocesos para explorar en mayor detalle [Molina & García, 2006].

Una vez *preprocesado* el conjunto de datos inicial y generado el conjunto de datos de entrenamiento asociado a cada clasificador binario, el siguiente paso es la construcción de estos clasificadores. Aunque, a priori, dichos clasificadores podrían construirse

usando cualquier método de clasificación supervisada (árboles de decisión, reglas de decisión, clasificadores *Bayesianos*...) en este trabajo se implementarán como *redes de neuronas* de una sola capa oculta entrenadas con el algoritmo de *Back-Propagation* [Rumelhart, et al., 1986]. Esta elección está condicionada por dos razones fundamentales:

1. Las redes de neuronas son modelos *inestables*. Es decir, los modelos obtenidos tras el proceso de aprendizaje están fuertemente condicionados por los datos usados en el entrenamiento y por el valor inicial de los pesos asociados a las interconexiones existentes entre las distintas neuronas. Por tanto, al variar el conjunto de datos de entrenamiento y/o el valor inicial asignado a los pesos es posible obtener modelos de clasificación distintos y, en consecuencia, aumenta la probabilidad de que sean diversos.
2. Gracias al masivo paralelismo de su estructura, se pueden implementar fácil y eficazmente sobre FPGAs [Ormondi & Rajapakse, 2006]. Esto permitirá que el modelo de clasificación diseñado para el *reconocimiento de señales de tráfico* (ver sección 5.6) pueda ser implementado como una solución *hardware* capaz de trabajar en tiempo real.

Así pues, cada clasificador binario se corresponde con una red de neuronas de una sola capa oculta en la que:

- a) El número de neuronas de entrada se corresponde con el número de características seleccionadas para el correspondiente problema binario.
- b) El número de neuronas ocultas se ha determinado experimentalmente en función de las características de cada dominio. Es decir, para cada dominio se han evaluado distintas arquitecturas de la capa oculta y se ha seleccionado la que, en términos de precisión y complejidad, mejor se ajusta al problema planteado.
- c) La capa de salida tiene una única neurona.

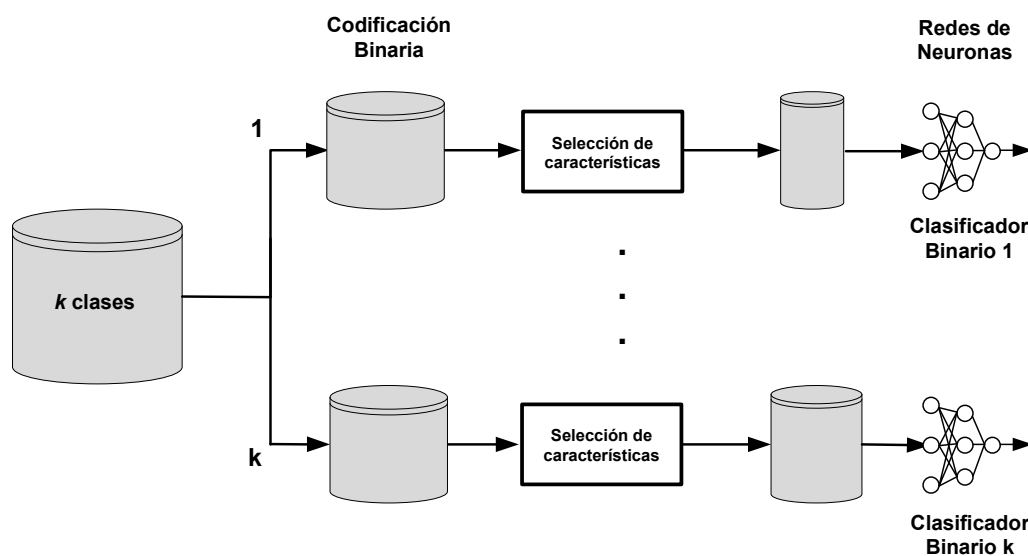


Figura 4.2. Construcción de los clasificadores binarios.

### Clasificadores complementarios

Atendiendo al esquema mostrado en la Figura 4.1, la función de los clasificadores complementarios,  $\overline{C}_i$ , es clasificar aquellos ejemplos que han sido rechazados (etiquetados como negativos) por el correspondiente clasificador binario ( $B_i$ ). Si el clasificador binario tiene un comportamiento ideal, estos ejemplos pertenecerán a una de las  $k-1$  clases que dicho clasificador aprende como negativas. Ante estas circunstancias, parece lógico admitir que los clasificadores complementarios deben construirse usando únicamente los ejemplos pertenecientes a las  $(k-1)$  clases que, en una situación ideal, deberá clasificar. Habitualmente los clasificadores no siguen un comportamiento ideal. Por tanto, aplicar esta filosofía implica que, ante un posible error del clasificador binario, el clasificador complementario se verá en el compromiso de tener que clasificar un ejemplo que no pertenece a ninguna de las clases que ha aprendido. A continuación se presenta un ejemplo con el que se pretende probar que esta circunstancia no supone una fuente de error para el conjunto.

Supongamos un dominio con cuatro clases  $\{c_1, c_2, c_3, c_4\}$  y admitamos que se ha construido un modelo de clasificación basado en la arquitectura BCE en el que la decisión final del sistema se obtiene por voto mayoritario. Admitamos también que, la salida dada por el módulo clasificador asociado a la  $i$ -ésima clase es un vector de  $k$  componentes ( $Y_i(\mathbf{x})=[y_1, y_2, \dots, y_k]$ ) en el que la componente  $y_i$  es generada por el clasificador binario y el resto de componentes por el clasificador complementario (ver apartado 4.2.2.a). Si los clasificadores que integran el conjunto son diversos y se asume que el ejemplo a clasificar pertenece a la clase  $c_1$ , las posibles situaciones que se pueden dar son las siguientes:

a) Todos los módulos base clasifican el ejemplo correctamente. En este caso, y tal y como se muestra en la figura, en el peor de los casos, la salida dada por la arquitectura BCE será  $Y(\mathbf{x})=\{4, 1, 1, 1\}$  por lo que la clase asignada al ejemplo será  $c_1$ . Por tanto, aunque la salida dada por el *módulo 1* sea ambigua (dos o más componentes de la salida toman valor 1), la salida dada por el conjunto es correcta.

	$y_1$	$y_2$	$y_3$	$y_4$
$M_1$	1	?	?	?
$M_2$	1	0	0	0
$M_3$	1	0	0	0
$M_4$	1	0	0	0
Votos	4	(1)	(1)	(1)

b) El clasificador binario del *módulo 1* comete un error: *Clasifica el ejemplo como negativo* (falso negativo). En este caso, aunque la salida dada por el *módulo 1* puede ser ambigua o incorrecta, la salida dada por el conjunto vuelve a ser correcta.

	$y_1$	$y_2$	$y_3$	$y_4$
$M_1$	0	?	?	?
$M_2$	1	0	0	0
$M_3$	1	0	0	0
$M_4$	1	0	0	0
Votos	3	(1)	(1)	(1)

c) El clasificador binario del *módulo 2* comete un error: *clasifica el ejemplo como positivo* (falso positivo). De nuevo, y a pesar de la posible ambigüedad de  $M_1$ , la salida del sistema es correcta.

	$y_1$	$y_2$	$y_3$	$y_4$
$M_1$	1	?	?	?
$M_2$	1	1	0	0
$M_3$	1	0	0	0
$M_4$	1	0	0	0
Votos	4	(2)	(1)	(1)

d) El clasificador complementario del *módulo 2* comete un error: *clasifica el ejemplo como perteneciente a la clase 3*. Bajo este supuesto, y aún forzando la situación más conflictiva ( $y_3(M_2)=1$ ), la salida dada por BCE es correcta.

	$y_1$	$y_2$	$y_3$	$y_4$
$M_1$	1	?	?	?
$M_2$	0	0	1	0
$M_3$	1	0	0	0
$M_4$	1	0	0	0
Votos	3	(1)	(2)	(1)

Del anterior análisis se puede concluir que con independencia de la salida dada por el clasificador complementario del *módulo 1*, en todos los casos y siempre que los clasificadores sean diversos, la salida dada por el conjunto es correcta. Es decir, desde un punto de vista teórico, el que un determinado clasificador complementario se vea en la tesitura de clasificar un ejemplo perteneciente a una clase que para él es desconocida no afecta al comportamiento del conjunto.

Además, si en el conjunto de datos usado para construir  $\overline{C_i}$  no se incluyen los ejemplos asociados a la clase  $c_i$  cada clasificador complementario se construirá usando un conjunto de entrenamiento distinto y, en consecuencia, aumenta la probabilidad de que dichos clasificadores sean diversos. Asimismo y dado que la heurística en la que se basa el método de selección de características considerado (CFS) depende tanto de los ejemplos como de sus clases, las variaciones realizadas en el conjunto de datos de entrenamiento derivan en subconjuntos de características distintos. Así, se puede concluir que cada clasificador complementario se construye usando un subconjunto de datos que se diferencia de los restantes en:

- i. Los ejemplos que lo integran (cada conjunto contiene unos determinados ejemplos).
- ii. Las características con las que se describen estos ejemplos.
- iii. La clase asignada a cada ejemplo.

Por tanto, es esperable que los clasificadores complementarios y, por ende, los módulos base que integran la arquitectura BCE, sean diversos.

La Figura 4.3 esquematiza el proceso seguido en la construcción de los clasificadores complementarios. Como se refleja en la Figura 4.3, y por los motivos mencionados en el epígrafe precedente, estos clasificadores serán implementados como una red de neuronas de una sola capa oculta y  $(k-1)$  neuronas en la capa de salida.

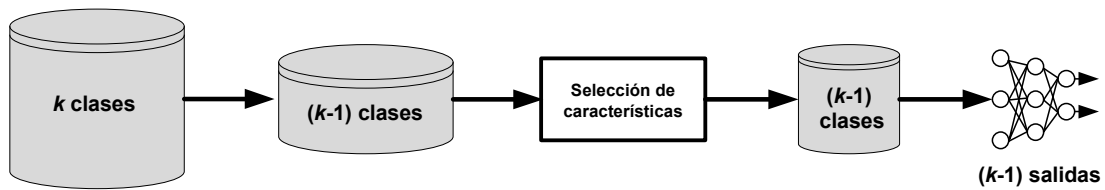


Figura 4.3. Construcción del clasificador complementario asociado a la  $i$ -ésima clase ( $\overline{C_i}$ ).

#### 4.2.2. Arquitectura de los módulos base: Integración Binario-Complementario.

Una vez contruidos los clasificadores que componen un módulo, el siguiente objetivo es determinar su interrelación. Con este fin, se proponen tres posibles opciones:

- a. *Combinación en paralelo.* Bajo esta perspectiva los clasificadores que componen un módulo base se consideran independientes y, por tanto, para conocer la salida asociada a un ejemplo es necesario conocer la salida generada tanto por el clasificador binario como por el complementario. Al aplicar esta arquitectura, la salida dada por el módulo clasificador asociado a la  $i$ -ésima clase será un vector de  $k$  componentes ( $Y_i(x)=[y_1, y_2, \dots, y_k]$ ) en el que el clasificador binario genera la componente  $y_i$  y el clasificador complementario el resto de componentes. Por tanto, para determinar la clase asociada a un ejemplo es necesario conocer la salida dada por ambos clasificadores.

- b. *Combinación en serie.* Bajo este esquema, la intervención del clasificador complementario está supeditada al valor de la salida del clasificador binario. Es decir, el clasificador complementario sólo interviene si el clasificador binario clasifica el ejemplo como no perteneciente a su clase. En este caso, la salida dada por el módulo base asociado a la  $i$ -ésima clase se corresponde con la descrita para la combinación en paralelo. En caso contrario, la salida de este módulo será un vector en el que la única componente distinta de cero será la generada por el clasificador binario  $C_i$ :  $Y_i(x)=[0, 0, \dots, y_i, \dots, 0]$ .

Puesto que la salida de los clasificadores binarios es un valor comprendido en el rango  $[0,1]$ , el punto crítico de este esquema pasa por determinar el umbral,  $\theta$ , que: i) permite, al clasificador binario, discernir si un ejemplos pertenece o no a su clase asociada ( $x \in c_i$  si y solo si  $y_i(x) > \theta$ ) y, ii) minimiza el error del conjunto. Por tanto, a priori, la elección de este umbral depende del dominio de aplicación.

- c. *Combinación jerárquica.* En este caso, la intervención de los clasificadores complementarios depende del resultado dado por los clasificadores que integran la *arquitectura OAA*. Es decir, se establece una dependencia jerárquica entre el conjunto formado por los clasificadores binarios (*arquitectura OAA*) y la actuación de los clasificadores complementarios. Así, si se considera que la *arquitectura OAA* es capaz de clasificar sin ningún tipo de ambigüedad un determinado ejemplo, dicho ejemplo no será enviado a ninguno de los clasificadores complementarios. Si, por el contrario, al evaluar las salidas dadas por los clasificadores dicotómicos el sistema no puede determinar la clase a la que pertenece un ejemplo, dicho ejemplo será enviado a los clasificadores complementarios.

Bajo esta perspectiva, se considerará que la *arquitectura OAA* clasifica un ejemplo sin ningún tipo de ambigüedad cuando las salidas  $y_i$  ( $i \in \{1, 2, \dots, k\}$ ) de los clasificadores binarios satisfacen la siguiente relación:

$$\exists i / y_i \geq \theta_1, \text{y, además, } y_j \leq \theta_2 \quad \forall j \neq i \quad [\text{Ec. 4.2}]$$

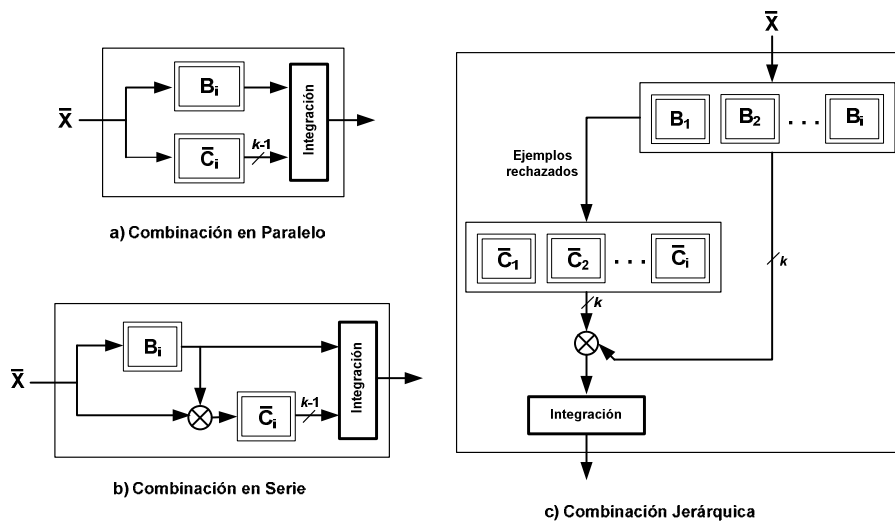


Figura 4.4. Integración Binario-Complementario: a) Paralelo; b) Serie; c) Jerárquica.

Aunque en un primer momento se pensó que la combinación en serie o la combinación jerárquica darían buenos resultados, el análisis experimental realizado sobre distintos dominios (ver Capítulo 5) ha mostrado que los mejores valores, en lo que a simplicidad y precisión se refiere, se obtienen cuando la arquitectura de *integración binario-complementario* sigue un esquema en paralelo [Sesmero, et al., 2012].

### 4.3. Descripción de CCE

Tras presentar y analizar las principales particularidades del sistema BCE, a continuación se propone un nuevo conjunto de clasificadores denominado CCE, *Complementary-Complementary Ensemble*, que pretende aumentar la precisión de las clasificaciones mediante el aumento de la diversidad de los módulos base.

El estudio de la Bibliografía nos indica que en el diseño de conjuntos de clasificadores *homogéneos* (conjuntos en los que los clasificadores base se construyen usando un único algoritmo de aprendizaje) la técnica predominante para generar clasificadores diversos es la manipulación del conjunto de datos de entrenamiento. Es decir, construir cada clasificador base usando un subconjunto de entrenamiento distinto. Aunque, a la hora de construir BCE se ha intentado que una de las fuentes de diversidad fuera la variación del subconjunto de datos de entrenamiento (ver epígrafe 4.2) esta técnica solo está explícita en la construcción de los clasificadores complementarios pero no en la construcción de los clasificadores binarios. En otras palabras, dadas las características de BCE:

- i) Todos los clasificadores binarios se construyen usando el mismo conjunto de datos de entrenamiento.
- ii) Admitiendo que el conjunto de datos de entrenamiento está *balanceado* (hay el mismo número de ejemplos de cada clase), la intersección de los conjuntos de entrenamiento usados en la construcción de dos complementarios cualesquiera tendrá una cardinalidad igual a  $\frac{(k-2)N}{k}$ , donde  $N$  es el número total de ejemplos disponibles y  $k$  el número de clases.

Según el anterior razonamiento, a medida que aumenta el número de clases, la diferencia porcentual entre los conjuntos de datos usados en la construcción de dos clasificadores complementarios cualesquiera disminuye y, por tanto, el grado de diversidad inducido, como consecuencia de la manipulación de los ejemplos usados en el entrenamiento, será menor.

Con el objetivo de garantizar que el grado de diversidad no depende del número de clases que describen un dominio, a continuación se propone una modificación del sistema BCE con la que se pretende superar esta limitación. Dado que, cada uno de los módulos base que integran este nuevo modelo ya no es la unión de un clasificador binario y un clasificador complementario sino de dos clasificadores complementarios entre sí, este nuevo modelo se ha denominado CCE (*Complementary-Complementary Ensemble*).

#### 4.3.1. Características de la arquitectura: Módulos base

Reiterando lo ya señalado, CCE surge como una extrapolación de BCE en la que cada módulo base está compuesto por dos clasificadores *multiclase* complementarios entre sí. Es decir, si el dominio sobre el que se trabaja tiene  $k$  clases, cada módulo estará formado por un clasificador entrenado con los ejemplos pertenecientes a  $j$  clases ( $1 < j < k$ ) y, el otro, con los ejemplos pertenecientes a las  $k-j$  clases restantes. Para intentar alcanzar el mayor grado de diversidad entre módulos, equilibrar los tiempos de entrenamiento requeridos para la construcción de cada módulo y, a su vez, reducir el número de posibles módulos, se impondrá, como una condición de diseño, que el valor de  $j$  ha de ser igual a  $\frac{k}{2}$ , si  $k$  es par o a  $\frac{k}{2}+1$  si  $k$  es impar. Con esta restricción, el número de posibles módulos pasa de:

$$2^{k-1} - (k + 1)$$

a:

$$\frac{1}{2} \frac{k!}{\left(\frac{k}{2}\right)!\left(\frac{k}{2}\right)!}, \text{ si } k \text{ es par}$$

ó

$$\frac{k!}{\left(\frac{k+1}{2}\right)!\left(\frac{k-1}{2}\right)!} = \frac{1}{2} \frac{(k+1)!}{\left(\frac{k+1}{2}\right)!\left(\frac{k+1}{2}\right)!}, \text{ si } k \text{ es impar}$$

Así, por ejemplo, si el número de clases de un determinado dominio es 6 el número de posibles módulos pasa de 25 a 10. Asumiendo que dichas clases son  $\{c_1, c_2, c_3, c_4, c_5, c_6\}$ , los posibles módulos que podemos construir tendrán la codificación de clases indicada en la Tabla 4.1.

**Tabla 4.1. Esquema de codificación de CCE para un problema de 6 clases.**

Módulo	Clasificador 1	Clasificador 2
<b>1</b>	$\{c_1, c_2, c_3\}$	$\{c_4, c_5, c_6\}$
<b>2</b>	$\{c_1, c_2, c_4\}$	$\{c_3, c_5, c_6\}$
<b>3</b>	$\{c_1, c_2, c_5\}$	$\{c_3, c_4, c_6\}$
<b>4</b>	$\{c_1, c_2, c_6\}$	$\{c_3, c_4, c_5\}$
<b>5</b>	$\{c_1, c_3, c_4\}$	$\{c_2, c_5, c_6\}$
<b>6</b>	$\{c_1, c_3, c_5\}$	$\{c_2, c_4, c_6\}$
<b>7</b>	$\{c_1, c_3, c_6\}$	$\{c_2, c_4, c_5\}$
<b>8</b>	$\{c_1, c_4, c_5\}$	$\{c_2, c_3, c_6\}$
<b>9</b>	$\{c_1, c_4, c_6\}$	$\{c_2, c_3, c_5\}$
<b>10</b>	$\{c_1, c_5, c_6\}$	$\{c_2, c_3, c_4\}$

Como se ha demostrado matemáticamente, la condición de diseño impuesta supone una considerable reducción en el número de posibles módulos y, por tanto, una reducción en el coste computacional del modelo. No obstante, y como queda recogido en la Tabla 4.2, para valores de  $k > 6$  el número de módulos que pueden integrar la arquitectura CCE es muy superior al de la arquitectura BCE.



**Tabla 4.2.** Número máximo de módulos base para los sistemas BCE y CCE en función del número de clases.

Núm. de Clases	Núm. de Módulos	
	BCE	CCE
<b>6</b>	6	10
<b>7</b>	7	35
<b>8</b>	8	35
<b>9</b>	9	126
<b>10</b>	10	126
<b>15</b>	15	6435
<b>20</b>	20	92378

Por tanto, al igual que sucede en otros conjuntos de clasificadores (p.e. *Bagging*, *Boosting*, ECOC o *Stacking*), consideramos que es necesario establecer un compromiso entre la posible mejora en la precisión del conjunto y el coste computacional que conlleva. Así, y basándonos en algunos estudios preliminares, impondremos, como una segunda condición de diseño, que el número de módulos que integran CCE no sea inferior a  $k$  pero tampoco superior a  $2k$ .

Una vez prefijado el número de módulos base de CCE, el siguiente objetivo es establecer la configuración de cada uno de ellos. Es decir, determinar las clases a partir de las cuales se han de construir los clasificadores que integran los distintos módulos.

Este proceso se podría realizar siguiendo la filosofía “*sobre-produce y elige*” [Gasmi & Merouani, 2005], es decir, generando todos los posibles módulos base y, seleccionando, con ayuda de alguna de las heurísticas indicadas en el epígrafe 2.2.2., el subconjunto que exhibe mayor precisión o mayor diversidad. Sin embargo, dado el coste computacional que lleva implícito este proceso, en la construcción de CCE se usará una técnica *ad hoc*. Es decir, previo a la construcción del conjunto se determinarán las clases específicas que servirán de base para la construcción de los clasificadores que integrarán el conjunto, y una vez prefijadas se construirán dichos módulos.

Por analogía con el método de representación usado en [Masulli & Valentini, 2000] para las arquitecturas de descomposición binaria, esta distribución de clases se representará como una matriz  $k \times M$  (donde  $M$  es el número de módulos establecido). En esta matriz, la componente  $x_{ij}$  indicará en cuál de los dos clasificadores que integran el  $j$ -ésimo módulo se consideran los ejemplos asociados a la  $i$ -ésima clase. Así, la matriz de códigos equivalente a la descomposición indicada en la Tabla 4.1, sería la recogida en la Tabla 4.3, donde, por ejemplo, en la primera línea se indica que las clases  $\{c_1, c_2, c_3\}$ , identificadas con un 0, se usarán en uno de los clasificadores del Módulo1 y las clases  $\{c_4, c_5, c_6\}$ , identificadas con un 1, en la construcción del otro clasificador.

**Tabla 4.3.** Matriz de descomposición de clases de CCE para un problema de 6 clases. En esta codificación los ceros representan las clases usadas en el primer clasificador complementario y los unos las clases usadas en el segundo clasificador complementario.

Módulo	$c_1$	$c_2$	$c_3$	$c_4$	$c_5$	$c_6$
$M_1$	0	0	0	1	1	1
$M_2$	0	0	1	0	1	1
$M_3$	0	0	1	1	0	1
$M_4$	0	0	1	1	1	0
$M_5$	0	1	0	0	1	1
$M_6$	0	1	0	1	0	1
$M_7$	0	1	0	1	1	0
$M_8$	0	1	1	0	0	1
$M_9$	0	1	1	0	1	0
$M_{10}$	0	1	1	1	0	0

El punto crítico de esta arquitectura pasa por determinar el mecanismo usado en la obtención de esta matriz de descomposición cuando el número de módulos es inferior a  $2^{k-1} - (k+1)$ . Por simplicidad, y aunque como trabajo futuro no descartamos el uso de técnicas basadas en aprendizaje no supervisado, en este trabajo la matriz de códigos se obtendrá de forma pseudo-aleatoria (ver epígrafe siguiente).

#### 4.3.2. Optimización de los módulos base

Al igual que sucede en la arquitectura OAO [García-Pedrajas & Haro, 2008], uno de los problemas que presenta la descomposición de clases propuesta para CCE es que, para cada patrón de entrada, existen varios clasificadores que están obligados a emitir una decisión errónea. Así, a la hora de clasificar un patrón de la clase  $c_i$ , todos los clasificadores que no han sido entrenados con instancias de esta clase, emitirán una decisión que siempre será errónea. Analizando la matriz de códigos reflejada en la Tabla 4.3, es fácil ver, que, dadas las características de CCE, esto sucede en uno de los dos clasificadores que integran cada módulo base. En una situación ideal en la que las clases son independientes cabe esperar que, al fusionar la salida de los dos clasificadores complementarios que integran un módulo, esta fuente de error sea subsanable. Es decir, se espera que la salida dada por el clasificador que sí ha aprendido esta clase, prevalezca sobre la salida dada por el clasificador para el que esta clase es desconocida. Sin embargo, en problemas en los que hay similitudes en las clases, esta circunstancia se puede convertir en una fuente de error cuyos efectos hay que minimizar. Así, por ejemplo, si  $c_1$  y  $c_4$  presentan grandes similitudes, todos aquellos módulos en los que  $c_1$  está asociado a uno de los clasificadores complementarios y  $c_4$  al otro, podrían, fácilmente, *confundir* los ejemplos de ambas clases. Bajo estas circunstancias, podría suceder que un gran número de ejemplos pertenecientes a  $c_4$  fueran calificados por el módulo como ejemplos de  $c_1$ . La experiencia nos sugiere que la única forma de minimizar este tipo de error es garantizando que ambas clases estén incluidas simultáneamente en el conjunto de entrenamiento usado en la construcción de, al

menos, un clasificador complementario. A continuación se analiza este aspecto en mayor detalle.

Supongamos una arquitectura CCE compuesta únicamente por los dos primeros módulos indicados en la Tabla 4.3. Admitamos que la salida de cada módulo viene dada por un vector  $Y_{Mi}=[y_1, y_2, \dots, y_6]$  en el que la componente  $y_i$  es generada por el primer clasificador del módulo  $M_i$  si y solo si la componente  $x_{ij}$  de la matriz de códigos es 0 y por el segundo clasificador si  $x_{ij}=1$ . Supongamos también, que cada componente del vector es un valor comprendido en el intervalo  $[0,1]$  y que, por tanto, se corresponde con la probabilidad de pertenencia a la clase. Y, por último, admitamos que la salida final del sistema se obtiene por promedio [Ec. 4.3], es decir:

$$C(\bar{x}) = \arg \max_{i=1}^k \left( \frac{\sum_{j=1}^M y_{ji}}{M} \right) \quad [\text{Ec. 4.3}]$$

donde:

$M$ =Número de módulos

$k$ = Número de clases

Si, tal y como se ha señalado,  $c_1$  y  $c_4$  presentan grandes similitudes se podría esperar que, a la hora de clasificar un ejemplo perteneciente a  $c_4$  la salida dada por el primer módulo base fuera:

$$Y_1=[0,9 \ 0,1 \ 0,1 \ 0,8 \ 0,1 \ 0,1]$$

Es decir, podría suceder que, para el primer módulo el ejemplo tuviera mayor probabilidad de pertenecer a la clase  $c_1$  que a la clase  $c_4$ . Sin embargo, para el segundo módulo, que posee un clasificador entrenado para distinguir, entre otras, las clases  $c_1$  y  $c_4$ , la situación, sería muy distinta y, por tanto, en ausencia de errores, la salida dada por este módulo podría ser del tipo:

$$Y_2=[0,3 \ 0,0 \ 0,1 \ 0,9 \ 0,2 \ 0,3]$$

Al combinar ambas salidas mediante la ecuación 4.3. la clase asignada por el sistema sería correcta. Es decir, el ejemplo sería clasificado como perteneciente a  $c_4$ .

El anterior análisis demuestra que una forma de subsanar los errores cometidos por aquellos clasificadores que no han aprendido una clase específica es garantizar que para dos clases cualesquiera, exista, al menos, un clasificador complementario entrenado con ejemplos de ambas clases.

Recapitulando lo indicado en este epígrafe se puede decir, que la arquitectura CCE es una arquitectura modular, que permite resolver problemas de clasificación en los que los ejemplos están descritos por  $k$  clases y en la que:

- i) El número de módulos está limitado, como máximo, a  $2k$ .
- ii) Cada módulo está compuesto por dos clasificadores *multiclase* complementarios entre sí, cada uno de los cuales se construye considerando únicamente los ejemplos pertenecientes a  $\frac{k}{2}$  clases.
- iii) Al establecer la descomposición de clases de los distintos módulos, se ha de garantizar que, para dos clases cualesquiera, exista, al menos, un clasificador complementario entrenado con ejemplos de ambas clases.

Además, y puesto que, uno de los objetivos que nos planteamos en esta Tesis Doctoral es la necesidad de resolver problemas en los que los ejemplos están descritos por un elevado número de características, al igual que en BCE, previo a la construcción de cualquier clasificador, se realizará un proceso de selección de características. En este proceso, solo se considerarán los ejemplos de entrenamiento usados en la construcción de cada clasificador y el método usado será algoritmo *BF+CFS* implementado en *WEKA*. Asimismo, y con el fin de garantizar que el modelo de clasificación resultante pueda ser implementado en *hardware* cada uno de los clasificadores base que integran la arquitectura serán redes de neuronas de una sola capa oculta.

En la Figura 4.5 se sintetiza el esquema de construcción de CCE.

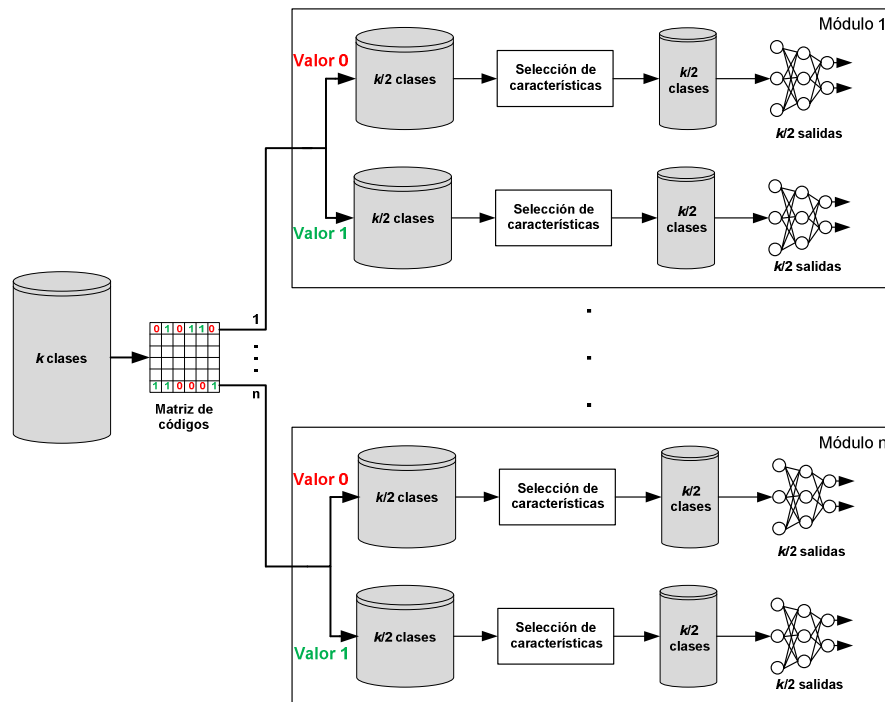


Figura 4.5. Esquema de construcción de CCE.

#### 4.4. Integración de los clasificadores base

Una vez descrito cómo se generan los módulos base que integran tanto BCE como CCE, en este epígrafe se completa la descripción arquitectónica de estos dos modelos indicado el método de integración usado para obtener la hipótesis final de cada conjunto.

Dado que los módulos base que integran cada conjunto se pueden considerar clasificadores que, de manera aislada, ofrecen una solución completa al problema planteado, el esquema de integración usado a la hora de determinar la decisión final del conjunto seguirá una arquitectura paralela. Además, la redundancia existente entre las decisiones individuales sugiere que la decisión final del sistema ha de obtenerse de forma cooperativa. Es decir, considerando la aportación de todos y cada uno de los módulos base. Bajo estas circunstancias, y según lo indicado en el epígrafe 2.3.1, la forma de obtener la decisión final del sistema es combinando las decisiones individuales mediante una función matemática o mediante un *metaclasificador*.

Por simplicidad, y dado que las salidas dadas por una red de neuronas se pueden interpretar como un indicativo de la probabilidad de pertenencia a las distintas clases, la decisión final de cada modelo se obtendrá promediando las salidas asociadas a cada clase y seleccionando la de mayor valor [Ec. 4.4], Es decir:

$$C(\bar{x}) = \arg \max_{i=1}^k \left( \frac{\sum_{j=1}^k y_{ji}}{M} \right) \quad [\text{Ec. 4.4}]$$

donde:

$y_{ji}$  = salida  $i$ -ésima del módulo clasificador asociado a la clase  $j$ .

$k$  = número de clases.

$M$  = número de módulos.

Una vez descritas las características arquitectónicas de los conjuntos de clasificadores propuestos en esta Tesis, en el siguiente capítulo se muestra el estudio de viabilidad realizado sobre ambos sistemas.



## Capítulo 5

### Validación Experimental

En el capítulo precedente se han presentado dos nuevos modelos de clasificación aplicables principalmente a dominios en los que el número de clases es finito y superior a dos y en los que los ejemplos están descritos por un gran número de atributos. Con el objetivo de determinar si, desde un punto de vista práctico, ambas propuestas permiten resolver eficazmente los problemas de clasificación para los que han sido concebidos, en este capítulo se presenta un análisis experimental en el que las arquitecturas descritas son testeadas sobre dominios de características dispares. En concreto, en este estudio experimental se pretende analizar la viabilidad de los métodos propuestos al considerar dominios con discrepancias en el número de clases, el número de atributos, el número de ejemplos de entrenamiento, la proporción de instancias pertenecientes a cada clase y/o el número de clases asignadas a cada ejemplo.

Para analizar la calidad de las arquitecturas propuestas, los resultados obtenidos en cada uno de los dominios han sido comparados con los derivados de aplicar algunos de los métodos de aprendizaje más usuales en la Bibliografía. Dado que los dominios usados en este proceso experimental tienen características dispares, la metodología de evaluación, análisis y comparación depende del dominio sobre el que se esté trabajando.

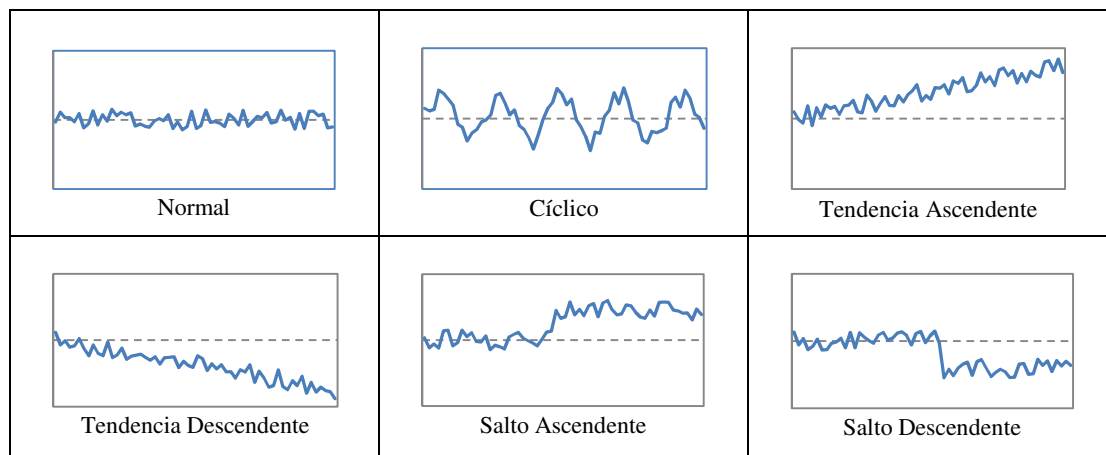
En los siguientes epígrafes se describen los dominios evaluados y se detalla el estudio experimental realizado en cada caso. Para facilitar la lectura, en la Tabla 5.1 se recogen las principales características de estos dominios.

**Tabla 5.1. Principales características de los dominios evaluados.**

Dominio	Ejemplos Entrenamiento	Ejemplos Test	Clases	Atributos	Desbalanceo	Multietiqueta
SYNTHETIC	300	300	6	60	No	No
SEMEION	796	796	10	256	No	No
SEMEION MODIFICADO	620	620	10	256	Sí	No
MNIST	60000	10000	10	784	No	No
REUTERS	23149	781265	101	47219	Sí	Sí
ASISTENTUR	450	450	9	1024	No	No

### 5.1. SYNTHETIC: Seis clases y conjunto de entrenamiento reducido

Para analizar la viabilidad de cada uno de los modelos propuestos sobre un dominio con pocas clases y pocos ejemplos de entrenamiento se ha optado por analizar las principales características funcionales de estos dos modelos sobre la base de datos SYNTHETIC [Frank & Asuncion, 2010]. Esta base de datos contiene 600 gráficos de control generados artificialmente y distribuidos equitativamente en una de seis clases (Figura 5.1): *normal*, *cíclico*, *tendencia ascendente*, *tendencia descendente*, *salto ascendente* y *salto descendente*. Cada ejemplo viene descrito por 60 características numéricas de valor variable (el valor de las distintas características está comprendido en el intervalo  $[-5,11; 63,83]$ ) por lo que, previo a la construcción de los modelos de clasificación, se ha realizado una normalización de cada atributo (el valor de cada atributo se ha normalizado al intervalo  $[0; 1]$ ).

**Figura 5.1. Ejemplos de la base de datos SYNTHETIC.**

A continuación se detalla el procedimiento experimental seguido al evaluar BCE y CCE sobre este dominio y se analizan los resultados obtenidos con cada uno de estos modelos.



### 5.1.1. Evaluación de BCE

Como se ha señalado en el capítulo precedente, la construcción de BCE conlleva la construcción de las redes de neuronas que integran la arquitectura (ver sección 4.2.1) y la elección del esquema de integración usado en los distintos módulos –esquema de integración Binario-Complementario– (ver sección 4.2.2). En los siguientes apartados se detallan estos aspectos y se muestran los resultados experimentales obtenidos al evaluar BCE sobre la base de datos SYNTHETIC.

### Arquitectura y Topología de las Redes de Neuronas

Aunque experimentalmente se ha demostrado que las redes de neuronas artificiales son modelos de clasificación especialmente útiles en ciertos problemas de clasificación (ejemplos incompletos, ruidosos, contradictorios o con una distribución no lineal) la obtención de la arquitectura y topología que mejor se ajusta a un determinado problema sigue siendo un tema abierto [Stathakis, 2009; Curteanu, et al., 2010]. Sin embargo, y dado que las dificultades que frente a la capacidad de generalización puede tener una red de neuronas se atenúan al combinarlas con otras redes [Zhang, 2000], resulta habitual que, al trabajar con conjuntos, esta búsqueda quede relegada a un segundo plano. Además, dado que el objetivo de la fase experimental presentada en este trabajo es analizar la viabilidad de las propuestas realizadas y estimar su calidad por comparación con otros modelos, parece lógico admitir que la elección de estos parámetros pueda realizarse de una manera estándar y homogénea.

Así, en este dominio, todas las redes de neuronas se han construido usando los siguientes valores:

- a) Topología. Se ha utilizado un *perceptrón* multicapa con una capa oculta y 15 neuronas en dicha capa. El número de neuronas en la capa de entrada viene determinado por el proceso de selección de atributos y el número de neuronas en la capa de salida por el tipo de red (binaria o *multiclase*).
- b) Función de activación de las neuronas ocultas y de salida. La función de activación utilizada es la función sigmoidea.
- c) Pesos de las conexiones. Los pesos de las conexiones se han inicializado de forma aleatoria y toman valores contenidos en el intervalo  $[-1, 1]$ .
- d) Algoritmo de aprendizaje. La actualización de pesos se realiza aplicando el algoritmo de *retropropagación* (*Back-Propagation*).
- e) Coeficiente de aprendizaje. El valor de este parámetro se ha fijado en 0,025.
- f) El número de ciclos en los que se desarrolla el aprendizaje se ha fijado en 300.

Como se puede observar en Figura 5.2, esta configuración parece garantizar una tasa de error inferior al 10% en todos los modelos considerados (arquitectura OAA implícita en BCE, clasificadores complementarios de BCE - $C_i$ - y red de neuronas simple).

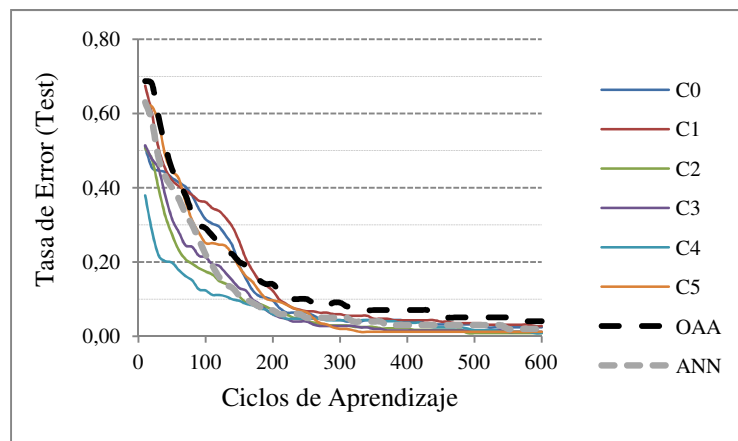


Figura 5.2. Tasa de error de los distintos clasificadores en función del número de ciclos de aprendizaje.

### Influencia del Esquema de Integración Binario-Complementario

El objetivo de este estudio es determinar el esquema de integración Binario-Complementario que ha de usarse en cada uno de los módulos base. Dado que el esquema de combinación en *paralelo* se puede interpretar como un caso particular tanto del *esquema en serie* ( $\theta=1$ ) como del *esquema jerárquico* ( $\theta_1=1$  ó  $\theta_2=0$ ), el análisis presentado en este epígrafe se limitará al estudio de estas dos arquitecturas de integración.

#### a) Combinación en Serie.

Bajo este esquema, la intervención de cada clasificador complementario está supeditada al valor de salida del correspondiente clasificador binario. Es decir, los clasificadores complementarios sólo intervienen cuando la salida,  $y_i(\mathbf{x})$ , dada por el correspondiente clasificador binario es menor que el umbral,  $\theta$ , previamente establecido. Por tanto, el objetivo abordado en este apartado es determinar si la precisión de BCE está o no influenciada por el valor de  $\theta$  y en caso afirmativo, establecer el mejor valor para este umbral.

Para garantizar la independencia de los resultados experimentales y evitar la influencia que sobre dichos resultados puede tener la elección del conjunto de entrenamiento y test, el tamaño de estos conjuntos y la aleatoriedad propia de las redes de neuronas (inicialización de los pesos) se ha optado por seguir el método experimental indicado en [Galindo et al., 2000] y que resumimos a continuación:

1. Dividir, aleatoriamente el conjunto de datos en dos particiones de igual tamaño.
2. Considerar una de estas particiones como ejemplos de entrenamiento y la otra como conjunto de test.
3. Generar 10 inicializaciones de pesos distintas y construir un modelo de clasificación a partir de cada una de estas inicializaciones.
4. Determinar el error de cada clasificador y excluir los casos extremos (modelos con mayor y menor error).
5. Intercambiar los conjuntos de entrenamiento y test y repetir los pasos 3 y 4.

6. Calcular el error medio del modelo.
7. Repetir el proceso anterior 30 veces.

Si se admite que un determinado valor de  $\theta$  da lugar a un modelo de clasificación distinto, para cada valor de  $\theta$  dispondremos de 30 valores de error (muestras) y aplicando el análisis de varianza de un factor (ANOVA) [Sheskin, 2000] podremos determinar si entre las medias de estos modelos (poblaciones) hay o no diferencias significativas. Es decir, podremos determinar si el esquema de integración en serie es o no invariable al valor de  $\theta$ .

La estrategia para poner a prueba la hipótesis de igualdad de medias consiste en obtener el valor del *estadístico F* mediante la ecuación 5.1:

$$F = \frac{\hat{\sigma}_1^2}{\hat{\sigma}_2^2} = \frac{\frac{b}{(b-1)} \sum (\bar{y}_i - \bar{y})^2}{\frac{1}{a-b} \sum \sum (y_{ij} - \bar{y}_i)^2} \quad [\text{Ec. 5.1}]$$

donde:

$\hat{\sigma}_1^2$ : es una estimación de la varianza poblacional basada en la variabilidad existente entre las medias de cada población.

$\hat{\sigma}_2^2$ : es una estimación de la varianza poblacional basada en la variabilidad existente dentro de cada grupo.

$y_{ij}$ : es el valor de la observación  $j$  dentro del grupo  $i$ .

$\bar{y}_i$ : es la media del grupo  $i$ .

$\bar{y}$ : es la media global.

$a$ : es el número de poblaciones.

$b$ : representa el número de muestras en cada población.

Bajo la *hipótesis nula* (las varianzas de todas las poblaciones son iguales), el estadístico  $F$  sigue aproximadamente una *distribución F* con  $a$  y  $(b-1)$  grados de libertad.

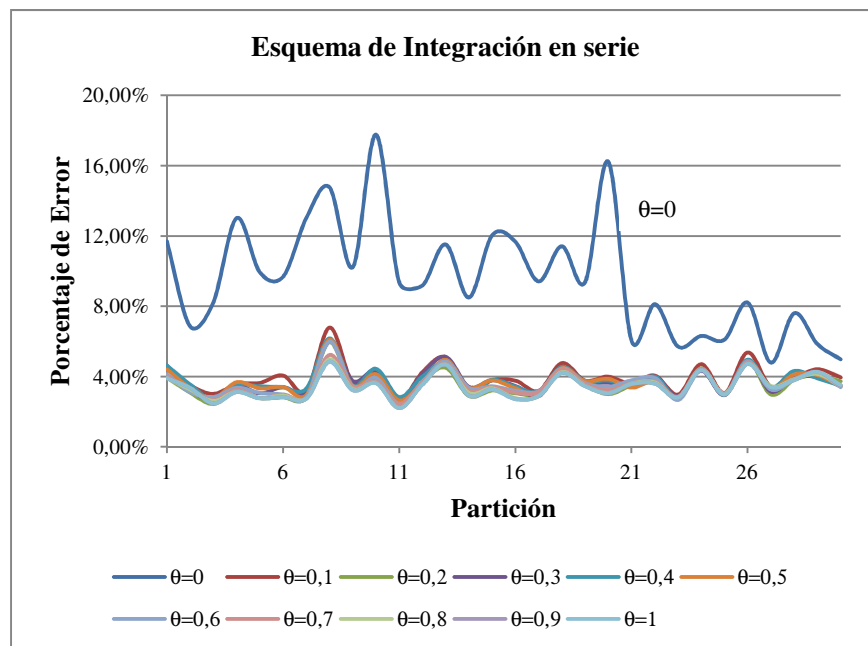
Por tanto, si el valor de  $F$  es menor que  $F_{0,05}(a, (b-1))$  -valor de la distribución  $F$  con  $a$  y  $(b-1)$  grados de libertad y un valor de significancia de 0,05- se podrá asegurar con un nivel de confianza del 95% que entre las medias de las poblaciones observadas no hay diferencias significativas.

Así, calculando el valor de  $F$  sobre los resultados derivados del proceso experimental indicado al comienzo de este epígrafe y comparándolo con el correspondiente valor crítico  $-F_{0,05}(a, (b-1))$ - podremos determinar si BCE, con un esquema de *integración Binario-Complementario en serie*, es o no estadísticamente invariable al valor de  $\theta$ .

En la Tabla 5.2 se muestra el porcentaje de error obtenido al aplicar este método experimental sobre la base de datos SYNTHETIC. La representación gráfica de estos valores se recoge en la Figura 5.3.

**Tabla 5.2. SYNTHETIC: Porcentaje de error en función de  $\theta$ . En las filas inferiores se indica la media, la mediana y la desviación estándar para cada población.**

Partición	$\theta=0$	$\theta=0,1$	$\theta=0,2$	$\theta=0,3$	$\theta=0,4$	$\theta=0,5$	$\theta=0,6$	$\theta=0,7$	$\theta=0,8$	$\theta=0,9$	$\theta=1$
1	0,117	0,044	0,039	0,046	0,046	0,044	0,041	0,040	0,039	0,039	0,039
2	0,069	0,035	0,031	0,033	0,035	0,034	0,031	0,033	0,033	0,034	0,034
3	0,082	0,030	0,024	0,026	0,027	0,028	0,028	0,026	0,026	0,025	0,025
4	0,130	0,036	0,032	0,035	0,035	0,037	0,034	0,033	0,031	0,031	0,031
5	0,099	0,036	0,028	0,030	0,034	0,033	0,031	0,028	0,028	0,028	0,028
6	0,097	0,041	0,028	0,034	0,034	0,034	0,030	0,030	0,029	0,028	0,028
7	0,130	0,031	0,028	0,031	0,033	0,030	0,028	0,028	0,028	0,028	0,028
8	0,148	0,068	0,060	0,061	0,062	0,061	0,060	0,052	0,050	0,049	0,049
9	0,102	0,037	0,033	0,038	0,035	0,035	0,034	0,035	0,033	0,032	0,032
10	0,178	0,043	0,038	0,043	0,044	0,041	0,039	0,037	0,036	0,036	0,036
11	0,094	0,026	0,024	0,026	0,028	0,026	0,024	0,024	0,022	0,022	0,022
12	0,092	0,043	0,037	0,042	0,040	0,037	0,037	0,037	0,036	0,036	0,036
13	0,115	0,051	0,045	0,051	0,048	0,050	0,049	0,047	0,047	0,047	0,047
14	0,085	0,031	0,029	0,034	0,030	0,034	0,034	0,032	0,031	0,029	0,029
15	0,120	0,038	0,032	0,038	0,038	0,038	0,034	0,034	0,033	0,033	0,033
16	0,117	0,038	0,030	0,035	0,034	0,033	0,032	0,031	0,028	0,027	0,027
17	0,094	0,032	0,030	0,029	0,031	0,030	0,032	0,031	0,029	0,029	0,029
18	0,114	0,048	0,042	0,045	0,045	0,045	0,044	0,045	0,043	0,042	0,042
19	0,094	0,038	0,035	0,037	0,037	0,038	0,036	0,036	0,035	0,035	0,035
20	0,162	0,040	0,030	0,036	0,038	0,039	0,034	0,032	0,031	0,031	0,031
21	0,060	0,036	0,035	0,035	0,034	0,034	0,038	0,036	0,036	0,036	0,036
22	0,081	0,041	0,036	0,040	0,040	0,037	0,039	0,037	0,037	0,036	0,036
23	0,057	0,030	0,028	0,029	0,028	0,028	0,027	0,029	0,028	0,028	0,028
24	0,063	0,047	0,045	0,043	0,044	0,044	0,044	0,044	0,044	0,044	0,044
25	0,061	0,031	0,030	0,029	0,030	0,030	0,030	0,030	0,030	0,030	0,030
26	0,082	0,054	0,049	0,049	0,050	0,048	0,049	0,048	0,047	0,047	0,047
27	0,048	0,034	0,030	0,031	0,034	0,033	0,033	0,034	0,035	0,034	0,034
28	0,076	0,039	0,038	0,040	0,043	0,041	0,038	0,039	0,038	0,038	0,038
29	0,059	0,044	0,041	0,040	0,039	0,041	0,042	0,042	0,041	0,043	0,043
30	0,050	0,040	0,037	0,035	0,035	0,035	0,034	0,035	0,035	0,035	0,035
Media	0,131	0,096	0,096	0,096	0,096	0,096	0,096	0,096	0,096	0,097	0,097
Mediana	0,131	0,095	0,096	0,096	0,096	0,096	0,096	0,096	0,096	0,096	0,097
Desv. Estándar	0,006	0,004	0,004	0,004	0,004	0,004	0,004	0,004	0,004	0,004	0,004



**Figura 5.3. SYNTHETIC: Porcentaje de Error en función de  $\theta$ .**

A partir de los valores mostrados en este gráfico se puede intuir que para valores de  $\theta \geq 0,1$  los sistemas resultantes son equivalentes. Para comprobar si desde el punto de vista estadístico esta conclusión es correcta, en la Tabla 5.3 se muestran los resultados obtenidos al realizar el análisis de varianza sobre los datos experimentales. Para facilitar la comprensión del estudio, el análisis estadístico se ha realizado: a) considerando los resultados obtenidos cuando  $\theta=0$  (primera fila) y b) omitiendo estos resultados (segunda fila).

**Tabla 5.3. SYNTHETIC: Análisis de varianza de un factor.**

Número de poblaciones	Número de muestras	$\hat{\sigma}_1^2$	$\hat{\sigma}_1^2$	$F$	Prob.	Valor crítico para $F$
11	330	0,00981	0,00015	<b>65,564</b>	5,12E-73	1,860
10	300	8,84E-05	5,48E-05	<b>1,61</b>	0,1115	1,912

Al comparar en cada uno de los casos el valor de  $F$  (*señalado en negrita*) con el correspondiente *valor crítico* (última columna), se corrobora que al incluir en la comparación el modelo obtenido para  $\theta=0$  (primera fila Tabla 5.3) las diferencias observadas son estadísticamente significativas ( $F > \text{Valor crítico}$ ). Por el contrario, al excluir el mencionado modelo (segunda fila Tabla 5.3) el análisis estadístico indica que las diferencias observadas no son significativas ( $F < \text{Valor crítico}$ ) y, por tanto, que los modelos analizados son estadísticamente equivalentes.

Dado que el modelo resultante para  $\theta=0$  se corresponde con la *arquitectura OAA* cuando ésta se combina con un proceso de selección de características (para  $\theta=0$  la intervención de los clasificadores complementarios queda anulada), y el modelo

resultante para  $\theta=1$  coincide con el esquema de *integración en paralelo* (los clasificadores complementarios intervienen en todos los casos), se puede concluir que:

- i. Cuando la *arquitectura OAA* se combina con un proceso de selección de características, el modelo resultante es menos preciso que BCE.
- ii. Para valores de  $\theta \geq 0,1$  la precisión de BCE es invariable al valor de  $\theta$  y la *combinación en serie* es estadísticamente equivalente a la *combinación en paralelo*.

b) Combinación jerárquica.

Para analizar si la *combinación jerárquica* ofrece ventajas sobre la *combinación en paralelo*, se ha seguido un método experimental similar al descrito en el epígrafe anterior con la salvedad de que, en este caso, cada modelo está determinado por los valores de  $\theta_1$  y  $\theta_2$  (ver epígrafe 4.2.2.c).

En la Figura 5.4 se representa el error medio de BCE en función del valor de los dos umbrales.

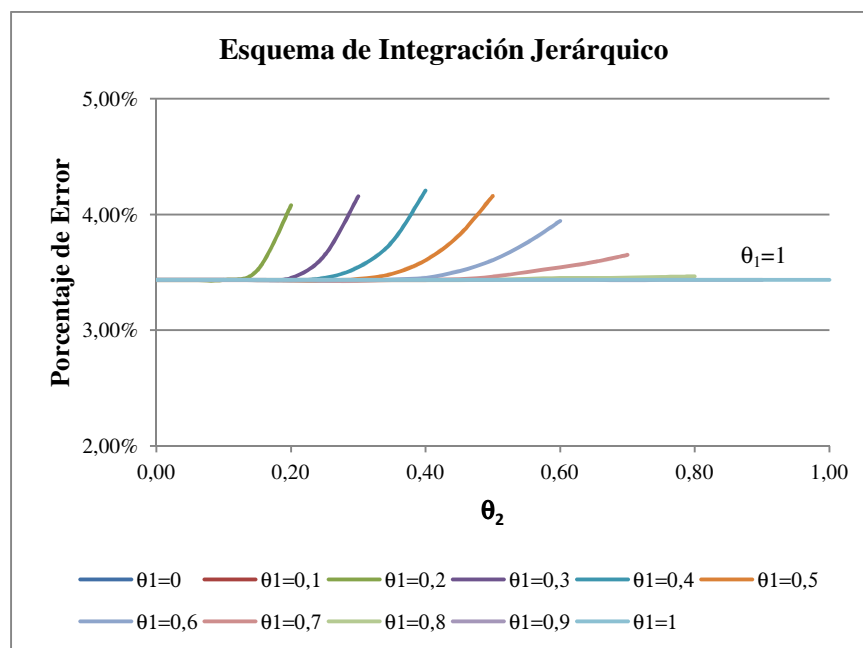


Figura 5.4. SYNTHETIC: Porcentaje de error para la combinación jerárquica.

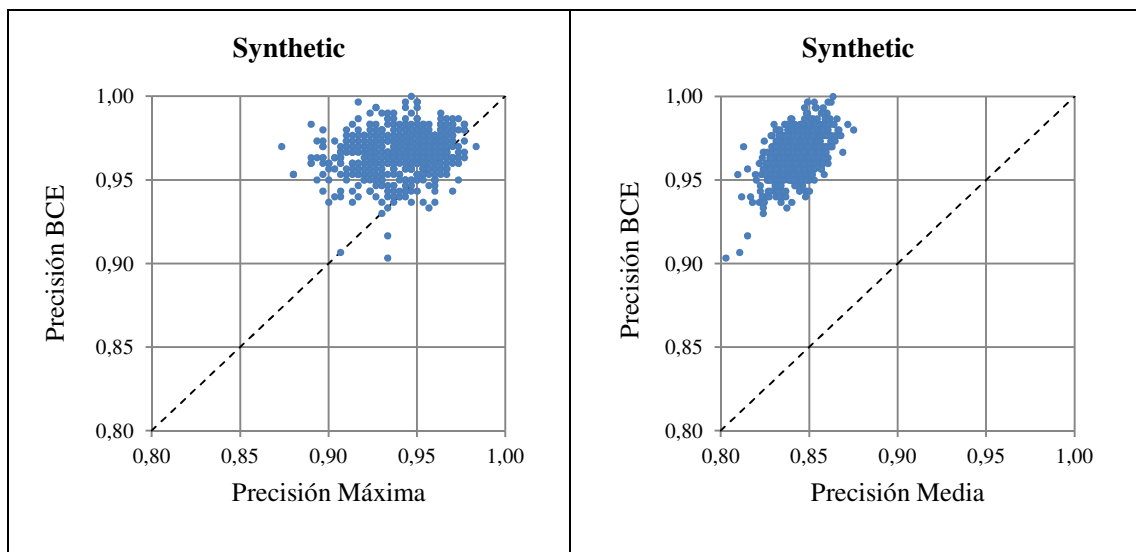
Observando la gráfica representada en la Figura 5.4 se puede comprobar que la *combinación jerárquica* nunca es mejor que la *combinación en paralelo* ( $\theta_1=1$  o  $\theta_2=0$ ).

Basándonos en los resultados de este estudio e imponiendo como máxima la simplicidad del diseño, en lo que sigue, el esquema de integración de los clasificadores que componen los módulos base de BCE seguirán un *esquema en paralelo* (ver Figura 4.4.a)

### Análisis de Precisión y Diversidad

Como se ha señalado en los anteriores capítulos, un conjunto de clasificadores es un grupo de clasificadores cuyas decisiones individuales se combinan con el objetivo de obtener un clasificador más preciso que cualquiera de sus miembros. Para lograr esta meta, es necesario que los clasificadores que integran el conjunto sean precisos y diversos.

Para comprobar si, en efecto, BCE es más preciso que cualquiera de sus miembros y, por tanto, verificar si la propuesta realizada tiene sentido en el marco de los conjuntos de clasificadores, se ha optado por comparar la precisión de BCE con a) la precisión del mejor módulo base y b) la precisión media de estos módulos. El resultado de este análisis queda recogido en la Figura 5.5. En dicha figura se representan las medidas relativas a todos y cada uno de los 600 (30x2x10) modelos de clasificación construidos.



**Figura 5.5. SYNTHETIC: Precisión Máxima y Precisión Media de los módulos base frente a la Precisión de BCE. La diagonal refleja los casos en los que la Precisión de BCE coincide con la Precisión Máxima o Media de sus módulos.**

Observando estas gráficas se puede comprobar que, en todos los casos analizados, la precisión obtenida con BCE es mayor que la precisión media de sus miembros, si bien BCE no siempre es más preciso que el mejor de sus módulos. Al realizar un análisis más detallado (consultar Apéndice A) se comprueba que en 521 de los 600 modelos construidos (87%) BCE es más preciso que cualquiera de sus miembros, mientras que la situación inversa solo se da en 51 ocasiones (9%). Además, más de la mitad de estas situaciones desfavorables (28 de 51) se concentran en 4 de las 30 particiones realizadas

(particiones 21, 24, 26 y 29) lo que sugiere que esta circunstancia adversa está más vinculada a las características de los datos usados en el entrenamiento/test que a las características de la arquitectura BCE.

Para determinar si la precisión de BCE es consecuencia de la diversidad inducida en los módulos base, a continuación se muestra, de forma gráfica, la relación existente entre la mejora media del conjunto ( $P_{BCE}-P_{MEDIA}$ ) y algunas de las medidas de diversidad detalladas en el Capítulo 2. Para facilitar la comprensión de este estudio, en la Tabla 5.4 se muestra un sumario de las medidas usadas y la relación teórica entre el valor de estas medidas y la precisión del conjunto.

**Tabla 5.4. Definición de algunas medidas de diversidad. El símbolo  $\uparrow/\downarrow$  indica que cuanto mayor es el valor de la medida, mayor/menor es el grado de diversidad.**

Nombre	Símbolo	Definición	$\uparrow/\downarrow$
Estadístico Q	Q	$\frac{N^{11}N^{00} - N^{01}N^{10}}{N^{11}N^{00} + N^{01}N^{10}}$	$\downarrow$
Coefficiente de correlación	$\rho$	$\frac{N^{11}N^{00} - N^{01}N^{10}}{\sqrt{(N^{11} + N^{10})(N^{01} + N^{00})(N^{11} + N^{01})(N^{10} + N^{00})}}$	$\downarrow$
Estadístico kappa	$\kappa$	$\frac{\sum_{i=1}^k N_{ii}}{N} - \sum_{i=1}^k \left( \frac{N_{i*}}{N} \frac{N_{*i}}{N} \right)$ $1 - \sum_{i=1}^k \left( \frac{N_{i*}}{N} \frac{N_{*i}}{N} \right)$	$\downarrow$
Medida de desacuerdo	des	$\frac{N^{01} + N^{10}}{N^{11} + N^{10} + N^{01} + N^{00}}$	$\uparrow$

donde:

$N$  es el número de ejemplos evaluados.

$L$  es el número de clasificadores base que integran el conjunto.

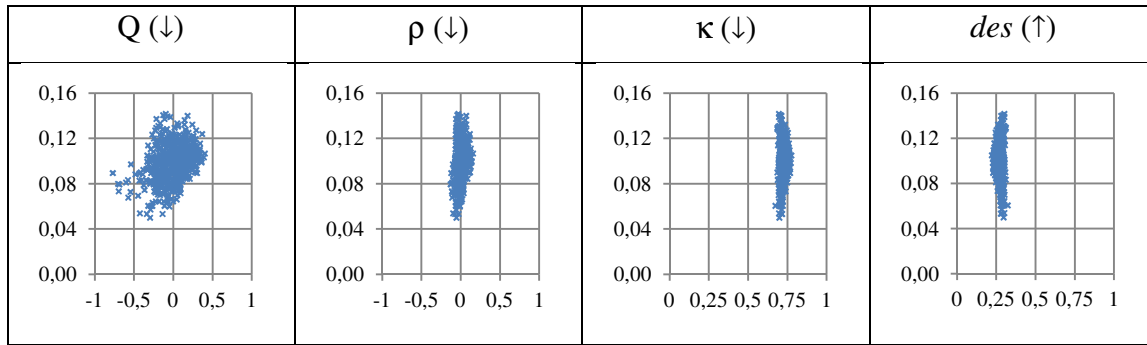
$K$  es el número de clases.

$N^{ab}$  es el número de ejemplos clasificados correctamente ( $a=1$ ) o incorrectamente ( $a=0$ ) por el  $i$ -ésimo clasificador base y correcta ( $b=1$ ) o incorrectamente ( $b=0$ ) por el  $j$ -ésimo clasificador base.

$N_{ij}$  es el número de ejemplos a los que el primer clasificador les ha asignado la clase  $i$  y el segundo clasificador les ha asignado la clase  $j$ .

La Figura 5.6 muestra, de forma gráfica, la relación entre las cuatro medidas de diversidad consideradas  $-Q, \rho, \kappa, des$  (eje de abscisas) y la mejora del conjunto (eje de ordenadas).



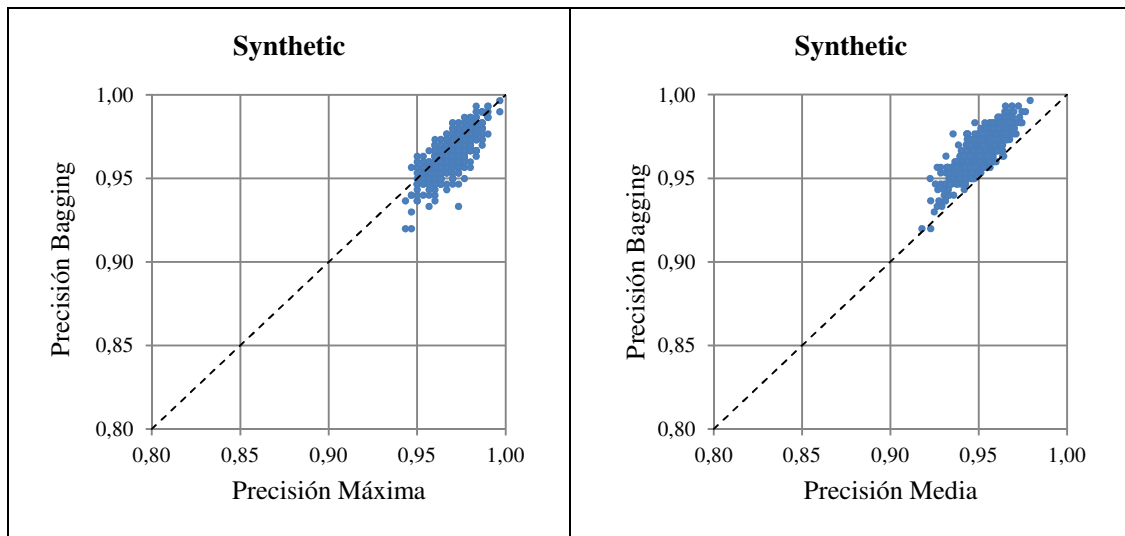


**Figura 5.6. Synthetic: Mejora de BCE ( $P_{BCE} - P_{MEDIA}$ ) frente a 4 medidas de diversidad ( $Q$ ,  $\rho$ ,  $\kappa$ ,  $des$ ). El símbolo  $\uparrow/\downarrow$  indica que cuanto mayor es el valor de la medida, mayor/menor es el grado de diversidad.**

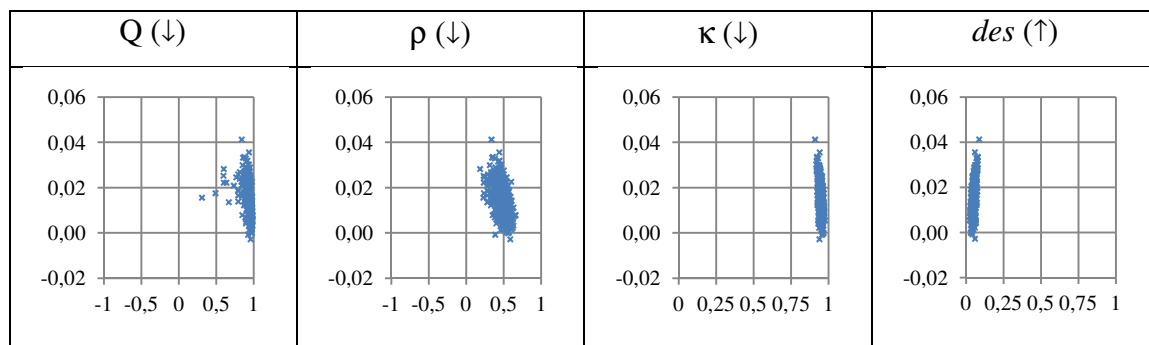
Observando las gráficas incluidas en la Figura 5.6 se puede intuir que la relación entre los valores teóricos y los observados depende de la métrica utilizada para cuantificar la diversidad. Así, esta relación es más acusada cuando el parámetro usado para medir la diversidad es  $Q$  o  $\rho$  y más débil (las curvas están desplazadas hacia el valor teórico más desfavorable) cuando el parámetro usado es  $\kappa$  o  $des$ . En lo que respecta a la relación entre las distintas medidas de diversidad y la precisión del conjunto, los resultados experimentales indican que no existe una relación directa entre ambos parámetros.

La escasa relación que parece existir entre los valores diversidad y la precisión de BCE podría interpretarse como un indicativo de debilidades en el modelo propuesto. Sin embargo, tras consultar la Bibliografía [Whitaker & Kuncheva, 2003; Kuncheva & Whitaker, 2001] y comprobar que aún no se ha podido demostrar que exista una relación directa entre diversidad y precisión, y que modelos como *Bagging* o *Boosting* pueden presentar valores de diversidad bastante alejados de los teóricos, concluimos que las observaciones expuestas en este epígrafe no tienen la suficiente solidez como para invalidar la propuesta realizada.

Para comprobar que, en efecto, las debilidades detectadas al evaluar BCE sobre este dominio están presentes en otros modelos de clasificación ampliamente valorados, seguidamente se muestran los resultados experimentales obtenidos al evaluar *Bagging* sobre esta base de datos. En concreto, en la Figura 5.7 se muestra la relación existente entre la precisión del conjunto y la precisión de los distintos clasificadores base (precisión del mejor clasificador y precisión media de estos clasificadores) y, en la Figura 5.8 la relación existente entre las medidas de diversidad calculadas y la mejora del conjunto. Para evitar la influencia que en los resultados puede tener la elección del algoritmo usado en la construcción de los clasificadores base, éstos se corresponden con redes de neuronas con una arquitectura y topología análoga a la usada en BCE. Es decir, *Bagging* ha sido implementado usando 10 redes de neuronas con 60 nodos de entrada, 15 nodos en la capa oculta y 6 nodos en la capa de salida, entrenadas durante 300 ciclos. Analizando los valores mostrados en la Figura 5.7, se puede comprobar que en el 64% de los casos, al aplicar *Bagging* sobre este dominio, la precisión obtenida por el conjunto es inferior a la del mejor de sus miembros (valores por debajo de la diagonal). Asimismo, al observar los valores recogidos en las gráficas de la Figura 5.8, se puede comprobar que la distancia entre los valores de diversidad obtenidos y los valores teóricos es más acusada en el caso de *Bagging* que en el caso de BCE.



**Figura 5.7. SYNTHETIC: Precisión Máxima/Media de los clasificadores base frente a la Precisión de Bagging.**



**Figura 5.8. SYNTHETIC: Mejora de la precisión de Bagging ( $P_{\text{BAGGING}} - P_{\text{MEDIA}}$ ) frente a las medidas de diversidad ( $Q$ ,  $\rho$ ,  $\kappa$ ,  $des$ ) calculadas.**

A la vista del estrecho margen de mejora que presentan tanto BCE como *Bagging* sobre el mejor de sus miembros (ver Figuras 5.5 y 5.7) se podría inducir que, sobre este dominio, el uso de un clasificador simple resulta más eficiente que la aplicación de un conjunto de clasificadores. Para refutar esta hipótesis, en la Tabla 5.5, se recogen los resultados experimentales obtenidos al evaluar BCE, *Bagging* y una red de neuronas simple (con igual arquitectura y topología que las usadas en *Bagging*) sobre esta base de datos así como el valor del *estadístico t pareado* obtenido al comparar la red de neuronas simple con *Bagging* y BCE respectivamente (última fila). Puesto que, los valores obtenidos para el *estadístico t* son mayores que el valor crítico correspondiente a un nivel de significancia de 0,05 (valor *t crítico* 2,045), se puede asegurar con un nivel de confianza del 95% que *Bagging* y BCE son estadísticamente mejores que el clasificador simple<sup>1</sup>.

<sup>1</sup> Al comparar *Bagging* y BCE mediante el *estadístico t* se concluye que ambos modelos son estadísticamente equivalentes ( $t=0,551$ ).

**Tabla 5.5. SYNTHETIC: Porcentaje de error medio obtenido al realizar 30 iteraciones del proceso de validación cruzada con 2 particiones. En negrita se muestran los mejores valores. En la última fila se recoge el valor del estadístico  $t$  obtenido al comparar la red de neuronas simple con *Bagging* y BCE.**

Iteración	ANN	Bagging	BCE
1	3,75%	<b>3,04%</b>	3,85%
2	3,88%	3,98%	<b>3,21%</b>
3	3,58%	3,10%	<b>2,44%</b>
4	<b>2,98%</b>	3,23%	3,12%
5	4,50%	3,06%	<b>2,83%</b>
6	3,85%	3,75%	<b>2,79%</b>
7	3,73%	3,67%	<b>2,79%</b>
8	3,92%	<b>3,62%</b>	4,88%
9	3,46%	<b>2,83%</b>	3,21%
10	3,60%	<b>3,40%</b>	3,60%
11	3,12%	3,06%	<b>2,27%</b>
12	3,88%	<b>3,19%</b>	3,48%
13	4,69%	<b>4,35%</b>	4,65%
14	3,71%	3,08%	<b>3,00%</b>
15	4,15%	3,48%	<b>3,27%</b>
16	3,33%	2,98%	<b>2,75%</b>
17	4,00%	3,21%	<b>2,92%</b>
18	3,35%	<b>3,15%</b>	4,00%
19	4,06%	3,50%	<b>3,46%</b>
20	4,10%	3,60%	<b>2,98%</b>
21	4,00%	3,56%	<b>3,54%</b>
22	4,15%	<b>3,56%</b>	3,69%
23	3,06%	3,42%	<b>2,81%</b>
24	4,38%	<b>3,40%</b>	4,33%
25	3,92%	3,25%	<b>2,90%</b>
26	4,44%	<b>4,10%</b>	4,73%
27	3,54%	3,46%	<b>3,40%</b>
28	4,23%	4,25%	<b>3,69%</b>
29	<b>3,71%</b>	4,25%	4,23%
30	3,94%	<b>3,44%</b>	3,46%
<b>Estadístico <math>t</math></b>		<b>4,913</b>	<b>3,900</b>

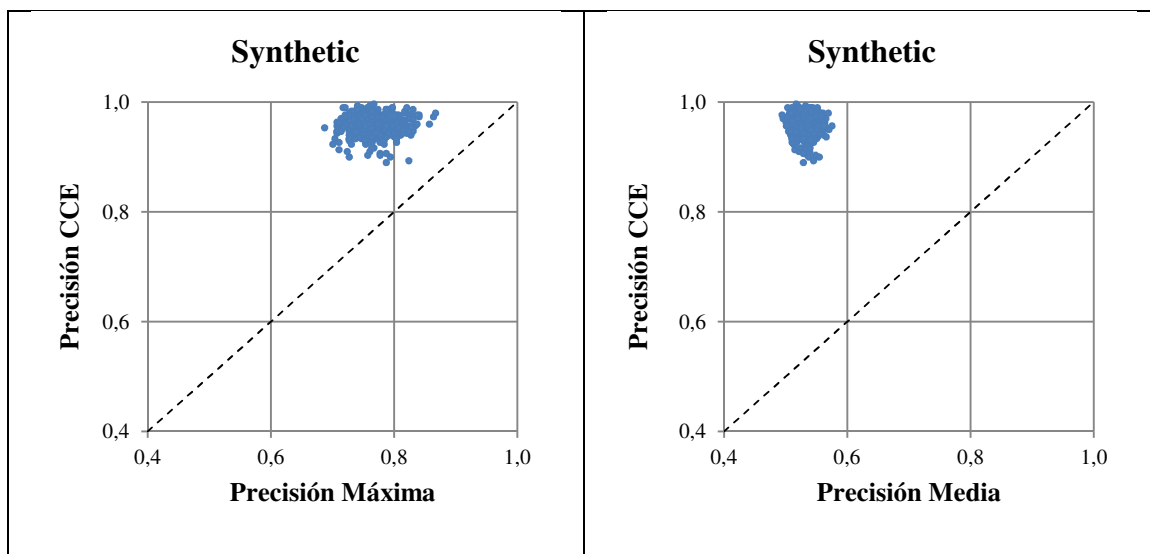
### 5.1.2. Evaluación de CCE

Una vez analizado el comportamiento de BCE sobre la base de datos SYNTHETIC, en este epígrafe se presentan los detalles del estudio realizado al evaluar dicha base de datos con CCE.

Dado que el número de clases contenidas en el dominio es 6, según lo expuesto en el epígrafe 4.3.1, cada uno de los clasificadores complementarios que componen un módulo base se entrenará con ejemplos pertenecientes a 3 de estas clases. Además, y dado que, para el número de clases indicado, el número máximo de módulos que se pueden construir es 10 y este número es inferior a  $2k$  (12), en este caso, CCE estará formado por todos los posibles módulos. Es decir, estará formado por 10 módulos y la matriz de descomposición de clases coincidirá con la indicada en la Tabla 4.3. Finalmente, señalar que, por analogía con las características arquitectónicas de BCE, los clasificadores usados en el diseño de CCE, serán redes de neuronas en las que el número

de nodos de la capa de entrada viene determinado por el proceso de selección de características y el número de neuronas de la capa oculta se ha fijado en 15. Además, para evitar sesgos atribuibles al método experimental, a la hora de determinar la precisión de este modelo, se aplicará el método experimental usado en la validación de BCE. Es decir, se realizarán 30 iteraciones del proceso de validación cruzada de 2 particiones garantizando que los datos de entrenamiento/test usados en cada iteración coinciden con los usados en la validación de BCE.

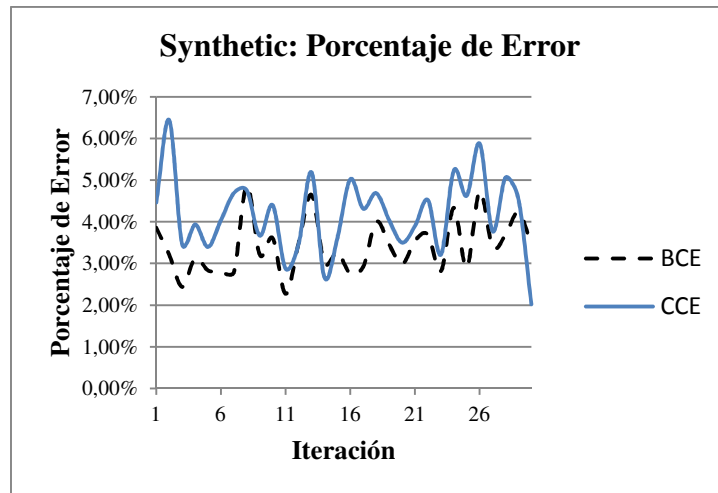
Para comprobar si, desde el punto de vista de la teoría de los conjuntos de clasificadores el modelo CCE es viable, es decir, para evaluar si la precisión del conjunto es mejor que la de cualquiera de sus miembros, en la Figura 5.9, se muestra la precisión de CCE frente a la precisión máxima/media de los módulos que lo integran.



**Figura 5.9. Synthetic: Precisión Máxima y Precisión Media de los módulos base frente a la Precisión de CCE. La diagonal indica los casos en los que  $P_{CCE}$  es igual a  $P_{MEDIA/MÁXIMA}$ .**

Observando los resultados mostrados en estas gráficas se puede comprobar que, en este caso y a diferencia de lo que sucede con el modelo BCE (ver Figura 5.5), la precisión del conjunto siempre es mayor que la precisión de cualquiera de sus miembros. Sin embargo, destaca la baja precisión que, en este caso, presentan los distintos módulos base respecto a la precisión total del conjunto ( $P_{MEDIA} \in [0,494;0,575]$ ;  $P_{CCE} \in [0,890;0,997]$ ). Esta circunstancia podría interpretarse como un indicativo de que, en este dominio, CCE no es la mejor alternativa. (Recordemos que la precisión de un conjunto de clasificadores suele depender tanto de la precisión como de la diversidad de los clasificadores base).

Para corroborar esta hipótesis, en la Figura 5.10 se recogen los valores de precisión medios obtenidos por CCE y BCE (esquema de integración en serie) en cada una de las iteraciones del proceso de validación cruzada.



**Figura 5.10.** Porcentaje de error al evaluar BCE y CCE (10 módulos base) sobre la base de datos SYNTHETIC.

Observando esta gráfica se puede comprobar que, en efecto, la precisión de BCE es mejor que la precisión obtenida por CCE. Para comprobar si estas diferencias son estadísticamente significativas, en la Tabla 5.6 se muestran los valores obtenidos al calcular el *estadístico t pareado* sobre los resultados experimentales mostrados en la Figura 5.10.

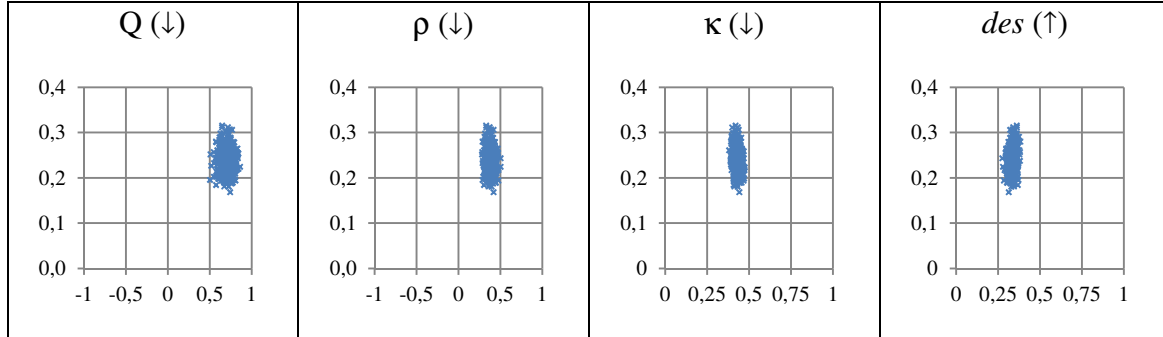
**Tabla 5.6.** SYNTHETIC: Comparación estadística (estadístico t pareado) de BCE y CCE. En **negrita** se muestra el valor obtenido para el estadístico t y el valor crítico para  $\alpha=0,05$ .

	BCE	CCE
Media	0,034	0,041
Varianza	4,441E-05	8,847E-05
Observaciones	30	30
Coef. de correlación de <i>Pearson</i>	0,498	
Diferencia hipotética de las medias	0	
Grados de libertad	29	
Estadístico t	<b>-5,015</b>	
P(T<=t) dos colas	2,423E-05	
Valor crítico de t (dos colas)	<b>2,045</b>	

Analizando los datos incluidos en la Tabla 5.6 se puede observar que el valor absoluto del *estadístico t* es superior al *valor crítico* correspondiente a un nivel de significancia de 0,05. Por tanto, se puede asegurar con un nivel de confianza del 95% que las diferencias observadas entre ambos modelos no se deben al azar y, en consecuencia, que, en este dominio concreto, *la arquitectura BCE es mejor que la arquitectura CCE*.

Para comprobar si la disminución en la precisión del conjunto está también condicionada por un descenso en el grado de diversidad de los módulos base, en la

Figura 5.11 se muestra, de forma gráfica, la relación entre la mejora del conjunto y las medidas de diversidad recogidas en la Tabla 5.4.



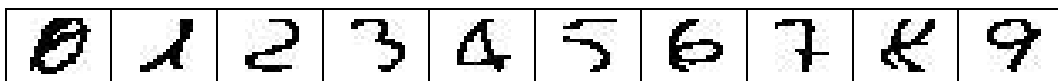
**Figura 5.11. SYNTHETIC: Mejora de la precisión de CCE ( $P_{CCE} - P_{MEDIA}$ ) frente a las medidas de diversidad ( $Q$ ,  $\rho$ ,  $\kappa$ ,  $des$ ) calculadas.**

Comparando los valores mostrados en la Figura 5.6 con los valores recogidos en la Figura 5.11 y recordando que el símbolo  $\uparrow/\downarrow$  indica que cuanto mayor es el valor de la medida mayor/menor es el grado de diversidad, es fácil concluir que, en este dominio, y según tres de las métricas usadas para cuantificar la diversidad ( $Q$ ,  $\rho$  y  $des$ ) los módulos base de CCE son menos diversos que los módulos base de BCE.

Recapitulando las distintas observaciones extraídas del estudio experimental se puede concluir que, al trabajar sobre la base de datos *SYNTHETIC*, la arquitectura BCE es mejor alternativa que la arquitectura CCE. Aunque esta conclusión resulta evidente cuando el parámetro de referencia es el valor de precisión de una y otra arquitectura, el que los módulos usados en BCE sean más precisos y diversos que los usados en CCE también se podría interpretar como un indicativo de la preeminencia de uno sobre otro.

## 5.2. SEMEION: Diez clases y pocos ejemplos de entrenamiento

Para analizar el comportamiento de BCE y CCE sobre un dominio con un número relativamente grande de clases y con no muchos ejemplos de entrenamiento, se ha seleccionado la base de datos *SEMEION*. Este conjunto de datos (ver Figura 5.12), está integrado por 1592 dígitos manuscritos provenientes de 80 personas distintas. Cada dígito está representado por 256 píxeles (16x16) que han sido normalizados a un valor binario (0 blanco / 1 negro). El número de clases es 10 (dígitos del 0 al 9) y el número de ejemplos por clases está distribuido de forma equitativa (159-160 ejemplos por clase).



**Figura 5.12. Ejemplos de la base de datos SEMEION.**

Dada la analogía que a nivel cuantitativo existe entre esta base de datos y la analizada en el epígrafe precedente (SYNTHETIC), el método experimental seguido en este estudio coincide con el método descrito anteriormente. Es decir, al igual que en el caso anterior, los valores experimentales mostrados se corresponden con los obtenidos al realizar 30 iteraciones del proceso de validación cruzada con dos particiones. Por otro lado, las características específicas de este modelo han condicionado que las redes de neuronas usadas en la construcción de este modelo contengan 20 neuronas en la capa oculta y que el entrenamiento se haya mantenido en 300 ciclos.

A continuación se muestran los resultados experimentales obtenidos sobre este dominio.

### 5.2.1. Evaluación de BCE

Como ya ha señalado a lo largo de este trabajo, la construcción de BCE implica determinar el esquema de integración *Binario-Complementario* usado en la construcción de los distintos módulos base. Por tanto, el punto de partida para la evaluación de BCE pasa por analizar las diferencias existentes entre los distintos esquemas de integración y determinar el esquema más apropiado. Dado que, el esquema de integración en paralelo está implícito tanto en la *arquitectura en serie* como en la *arquitectura en paralelo*, a continuación se resume el análisis realizado al considerar éstos dos últimos esquemas.

En la Figura 5.13 se muestra el error de BCE en función del valor del umbral ( $\theta$ ) establecido para la *combinación en serie*.

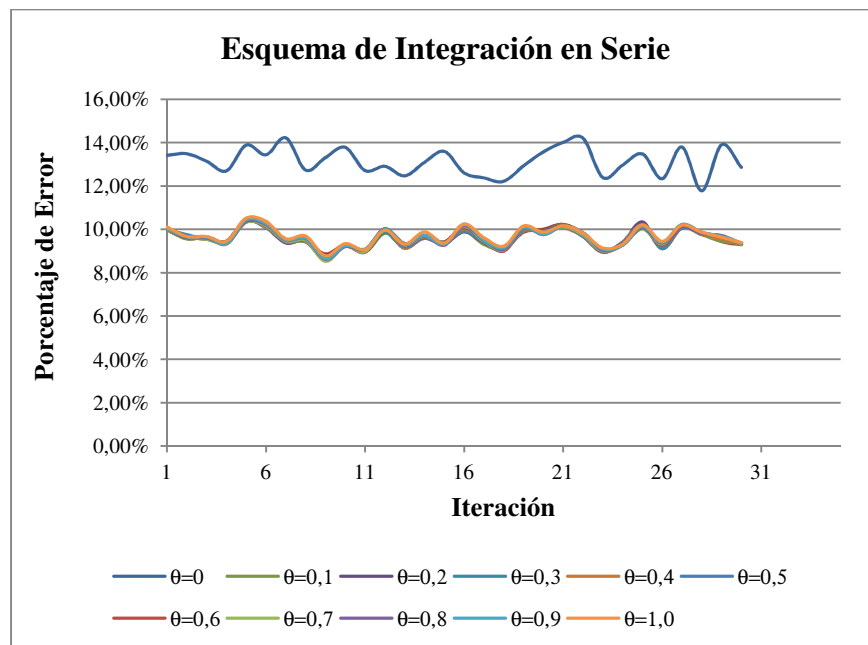


Figura 5.13. SEMEION: Combinación en serie. Porcentaje de Error en función de  $\theta$ .

A partir de los valores mostrados en la Figura 5.13 se puede comprobar que, al igual que sucede en la base de datos SYNTHETIC, el único valor de  $\theta$  para el que el sistema parece

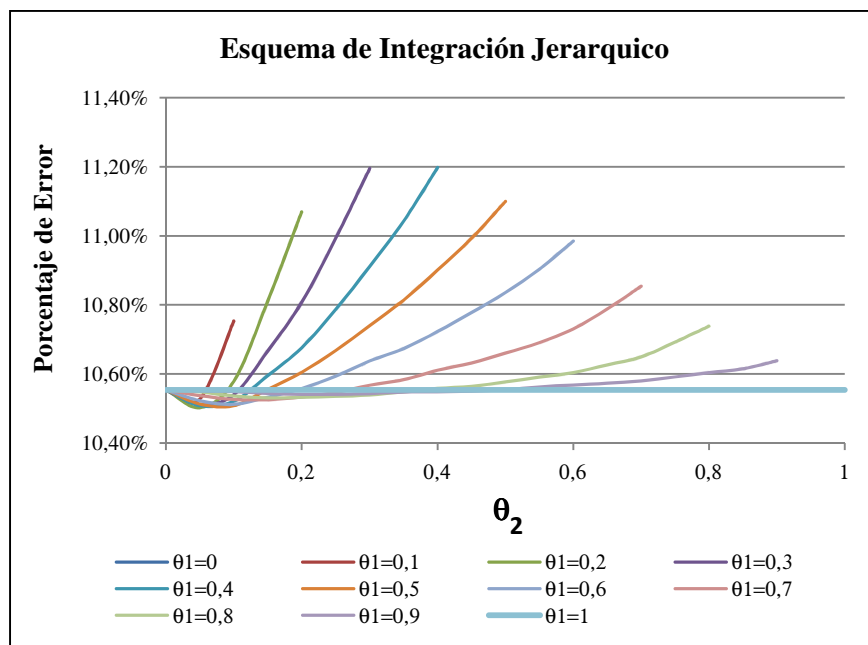
tener un peor comportamiento es  $\theta=0$ . Para comprobar si estas diferencias son o no estadísticamente significativas, en la Tabla 5.7 se recogen los valores obtenidos al realizar el *análisis de varianza de un factor* considerando los valores relativos a  $\theta=0$  (primera fila) y excluyéndolos (segunda fila).

**Tabla 5.7. SEMEION: Análisis de varianza de un factor.**

Número de poblaciones	Número de muestras	$\hat{\sigma}_1^2$	$\hat{\sigma}_1^2$	$F$	Prob.	Valor crítico para $F$
11	330	0,00340	2,06E-05	<b>164,89</b>	1,3E-119	1,860
10	300	4,12E-06	1,85E-05	<b>0,222</b>	0,991	1,912

Al comparar, en cada uno de los casos, el valor del *estadístico  $F$*  (señalado en negrita) con el correspondiente valor crítico se corrobora, que mientras en el primer caso las diferencias observadas son estadísticamente significativas ( $F > \text{Valor crítico}$ ) en el segundo no lo son. Por tanto, de nuevo, se puede concluir que para valores de  $\theta \geq 0,1$  la precisión de BCE es invariable al valor de  $\theta$  y la *combinación en serie* es estadísticamente equivalente a la *combinación en paralelo* ( $\theta=1$ ).

Una vez analizado el esquema de *integración en serie*, en la Figura 5.14 se muestra el porcentaje de error obtenido al analizar el esquema de *integración jerárquico*.



**Figura 5.14. SEMEION: Porcentaje de error para la combinación jerárquica.**

Observando estas gráficas se puede comprobar que, en la mayoría de los casos, la combinación jerárquica es peor que la combinación en paralelo ( $\theta_1=1$  o  $\theta_2=0$ ). Para



comprobar si, en algún caso, la combinación jerárquica es estadísticamente mejor que la combinación en paralelo, se ha optado por realizar el análisis de varianza sobre aquellos modelos que tienen un error medio menor al obtenido con la combinación en paralelo. Los resultados de este análisis se recogen en la Tabla 5.8.

**Tabla 5.8. SEMEION. Análisis de Varianza: Combinación jerárquica.**

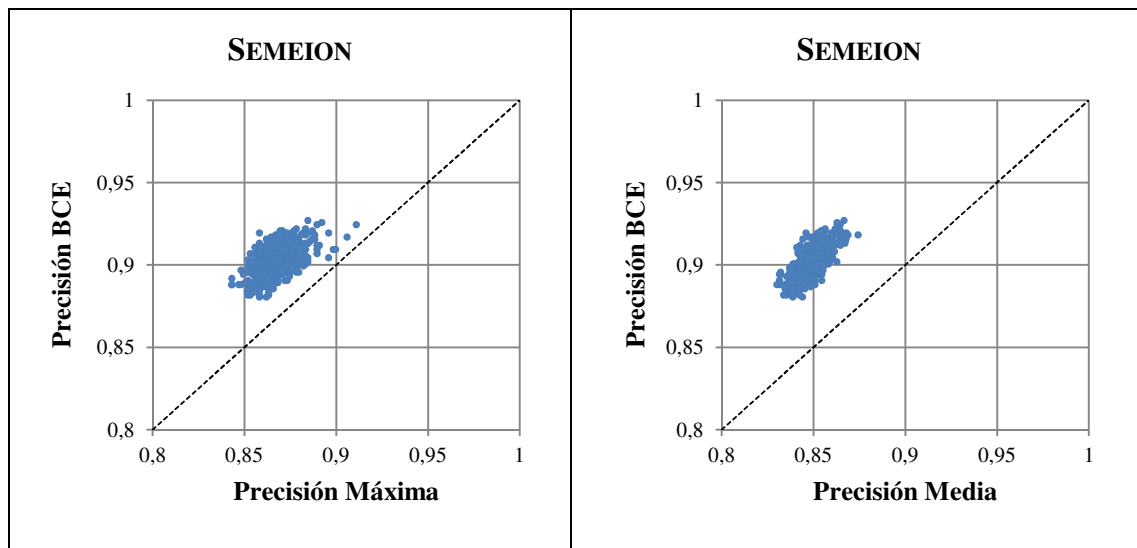
Número de Poblaciones	Número de Muestras	$\hat{\sigma}_1^2$	$\hat{\sigma}_1^2$	$F$	Prob,	Valor crítico para $F$
32	30	0,383	9,969	<b>0,038</b>	1	1,464

Comparando, el valor obtenido para el *estadístico F* (señalado en negrita) con el correspondiente *valor crítico*, se concluye que entre los modelos analizados no hay diferencias significativas ( $F < \text{Valor Crítico}$ ) y, por tanto, que incluso ante los mejores valores de  $\theta_1$  y  $\theta_2$ , la *combinación jerárquica* es equivalente a la *combinación en paralelo*.

Así pues, dada la equivalencia estadística existente entre los distintos esquemas de integración, en lo que sigue y por simplicidad, la arquitectura de integración Binario-Complementario implícita en BCE será el *esquema en paralelo*.

### Análisis de Precisión y Diversidad

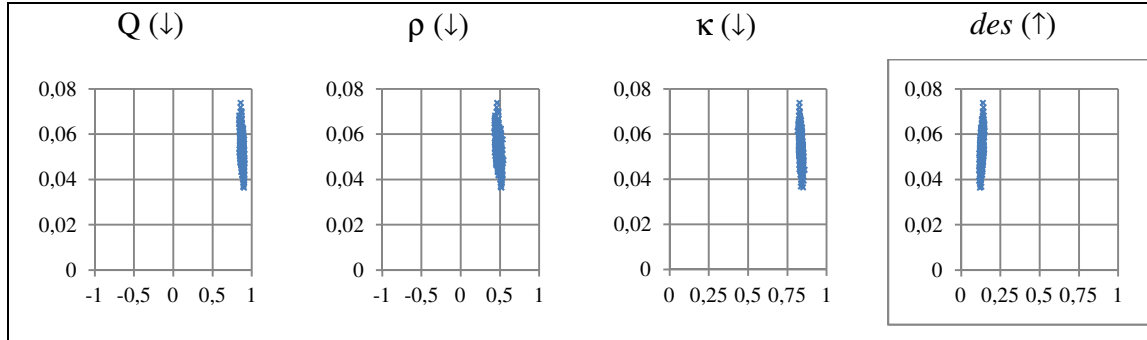
Para comprobar si, sobre la base de datos SEMEION, se satisface que BCE es mejor que cualquiera de sus miembros, en la Figura 5.15 se muestra el valor de precisión del conjunto frente a la precisión máxima/media de los módulos que lo integran.



**Figura 5.15. SEMEION: Precisión Máxima y Precisión Media de los módulos base frente a la Precisión de BCE.**

Observando estas gráficas se puede comprobar que, en este caso, el conjunto siempre es más preciso que el mejor de sus módulos y, por tanto, mejor que cualquiera de ellos.

Para determinar si, en este caso, la mejora lograda en la precisión del conjunto es consecuencia de la diversidad inducida, a continuación, en la Figura 5.16, se muestra la relación existente entre ambos parámetros cuando las métricas usadas para cuantificar la diversidad son las recogidas en la Tabla 5.4.



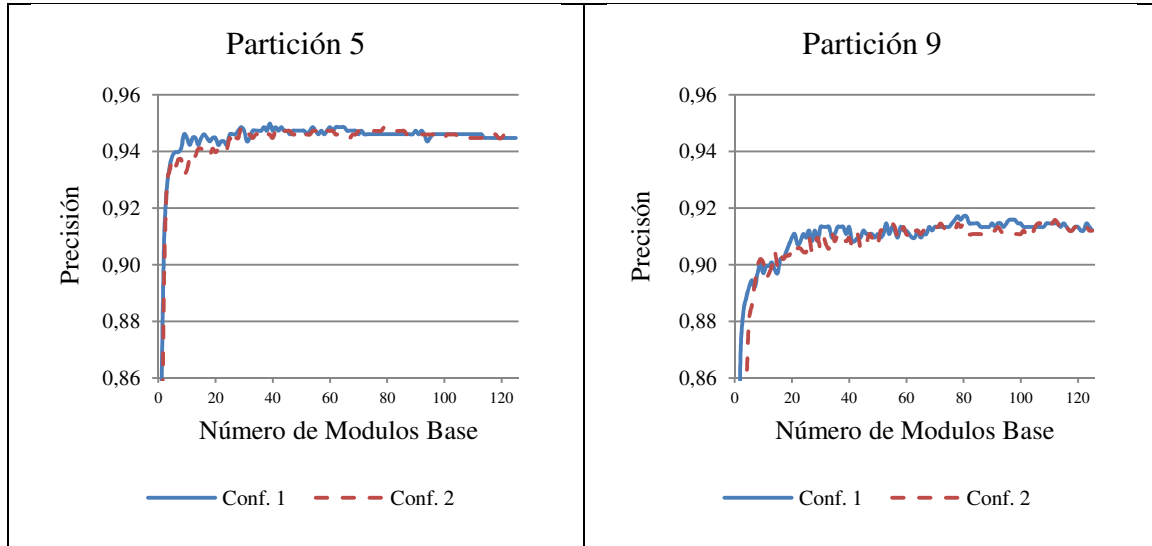
**Figura 5.16. Semeion: Mejora de BCE (PBCE-PMEDIA) frente a 4 medidas de diversidad ( $Q$ ,  $\rho$ ,  $\kappa$ ,  $des$ ). El símbolo  $\uparrow/\downarrow$  indica que cuanto mayor es el valor de la medida, mayor/menor es el grado de diversidad.**

Observando las gráficas incluidas en la Figura 5.16 se puede comprobar que la relación existente entre los valores de diversidad y la mejora del conjunto es bastante difusa. Así, por ejemplo un valor de  $Q=0,86$  implica, dependiendo del caso, una mejora en la precisión ( $P_{BCE}-P_{MEDIA}$ ) que oscila entre 0,043 y 0,074. Además, se puede observar que los modelos construidos a partir de la base de datos SEMEION tienen valores de diversidad muy alejados de los valores teóricos esperados (altos cuando se espera un valor bajo  $-Q$ ,  $\rho$ ,  $\kappa$  y bajo cuando se espera un valor alto  $-des$ ). Dado que, al evaluar BCE sobre esta base de datos el conjunto siempre es más preciso que cualquiera de sus miembros, se esperaba que al cuantificar la diversidad, los valores obtenidos estuvieran más próximos a los valores teóricos. La diferencia entre estos valores nos hace intuir que existe un rango de mejora y, por tanto, que si logramos inducir un mayor grado de diversidad en los clasificadores base se logrará un aumento en la precisión del conjunto. Con el fin de lograr este objetivo, y puesto que la arquitectura CCE premia la diversidad entre los módulos base frente a la precisión de éstos, a continuación, se presentan los valores experimentales obtenidos al evaluar CCE sobre esta base de datos.

### 5.2.2. Evaluación de CCE

Dado que, en el caso que nos ocupa, la *cardinalidad* del conjunto de clases es 10, según lo indicado en el epígrafe 4.3.1, CCE estará compuesto por entre 10 y 20 módulos, cada uno de los cuales estará formado por dos clasificadores quinararios. Puesto que, dadas 10 clases el número total de módulos que pueden constituir la arquitectura CCE es 126, en primer lugar analizaremos la relación que, en este dominio, existe entre la precisión de CCE y el número de módulos base. Para realizar este análisis se han seleccionado como conjuntos de entrenamiento y test aquellas particiones para las que BCE genera el mayor/menor porcentaje de error (ver Figura 5.13). Para cada una de estas particiones –particiones 5 y 9– se han construido los 126 módulos base y una vez creados se han integrado en la arquitectura CCE de forma totalmente aleatoria. Para comprobar que los

resultados son independientes del orden establecido, el proceso anterior se ha repetido dos veces. Los resultados de este análisis quedan recogidos en la Figura 5.17.

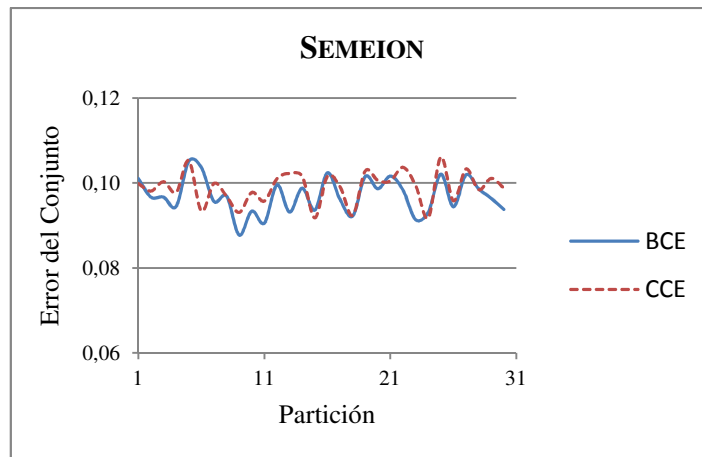


**Figura 5.17. SEMEION: Precisión de CCE en función del número de módulos integrados en la arquitectura.**

Analizando las gráficas contenidas en la Figura 5.17 se observa que tras incluir los 10 primeros módulos base, la precisión del sistema queda prácticamente estabilizada. Por tanto, admitiremos que la mejor configuración, en términos de precisión/complejidad computacional, se logra usando 10 módulos base.

Una vez prefijada la arquitectura de CCE para este dominio, el siguiente objetivo es determinar la topología de cada uno de sus miembros. Es decir, determinar la configuración específica de clases de cada uno de los módulos que formarán parte del modelo. Establecer la topología de estos módulos se puede interpretar como la selección de un parámetro variable del modelo y, en consecuencia, como la selección de un parámetro que puede tener influencia en los resultados experimentales. Por ello, y con el objetivo de garantizar la independencia de la evaluación, cada uno de los 600 modelos (30x2x10) en los que deriva el método experimental expuesto en el epígrafe 5.1.1 tendrá, en la medida de lo posible, una configuración de clases específica. Según lo especificado en los apartados 4.3.1 y 4.3.2, esta configuración se obtendrá de forma aleatoria pero garantizando que para dos clases cualesquiera siempre ha de existir, al menos, un clasificador complementario entrenado con ejemplos de ambas clases.

En la Figura 5.18 se muestran los valores de precisión medios obtenidos por CCE (10 módulos base) y BCE (esquema de integración en serie) en cada una de las iteraciones del proceso de validación cruzada. En ambos casos, cada módulo base está formado por dos redes de neuronas con 20 nodos ocultos entrenadas durante 300 ciclos.



**Figura 5.18.** Porcentaje de error al evaluar BCE y CCE (10 módulos base) sobre la base de datos SEMEION.

Como se puede comprobar al comparar las gráficas contenidas en la Figura 5.18, en este caso, los resultados obtenidos con ambos sistemas parecen equivalentes. Sin embargo, al calcular el *estadístico t* sobre estos valores (Tabla 5.9) se comprueba que las diferencias observadas son estadísticamente significativas ( $|t| > t_{0,05;29}$ ). Analizando el error medio de cada sistema se deduce que, de nuevo, el sistema que parece tener un mejor comportamiento es BCE (posee un error medio menor).

**Tabla 5.9.** SEMEION: comparación estadística (*prueba t pareada*) de BCE y CCE. En negrita se muestra el valor obtenido para el *estadístico t* y el valor crítico de *t* para  $\alpha=0,05$ .

	BCE	CCE
Media	0.097	0.099
Varianza	1.830E-05	1.614E-05
Observaciones	30	30
Coeficiente de correlación de Pearson	0.606	
Diferencia hipotética de las medias	0	
Grados de libertad	29	
Estadístico t	<b>-3.025</b>	
P(T<=t) dos colas	0,005	
Valor crítico de t (dos colas)	<b>2.045</b>	

Al analizar los valores tanto de precisión (Figura 5.19) como de diversidad (Figura 5.20) de los módulos base frente a la precisión del conjunto se verifica que CCE siempre es más preciso que cualquiera de sus miembros y que éstos son más diversos (las gráficas están ligeramente desplazadas hacia el mejor valor teórico) que los clasificadores que integran BCE. Es más, al cuantificar la diversidad y establecer en términos del coeficiente de correlación de *Pearson* [Spiegel, 1976] (Ec. 5.2) y del coeficiente de correlación de *Spearman* [Spiegel, 1976] (Ec. 5.3) la relación existente entre este parámetro y la mejora del conjunto ( $P_{CJTO} - P_{MEDIA}$ ) - Tabla 5.9 - se comprueba que esta relación es más fuerte en CCE que en BCE.

$$\rho = 1 - \frac{6 \sum d^2}{n(n^2-1)} \quad [\text{Ec. 5.2}]$$

$$r_{xy} = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n \sum x_i^2 - (\sum x_i)^2} \sqrt{n \sum y_i^2 - (\sum y_i)^2}} \quad [\text{Ec. 5.3}]$$

donde:

$n$ : número de valores.

$d$ : es la diferencia entre los correspondientes estadísticos de orden  $x$ - $y$ .

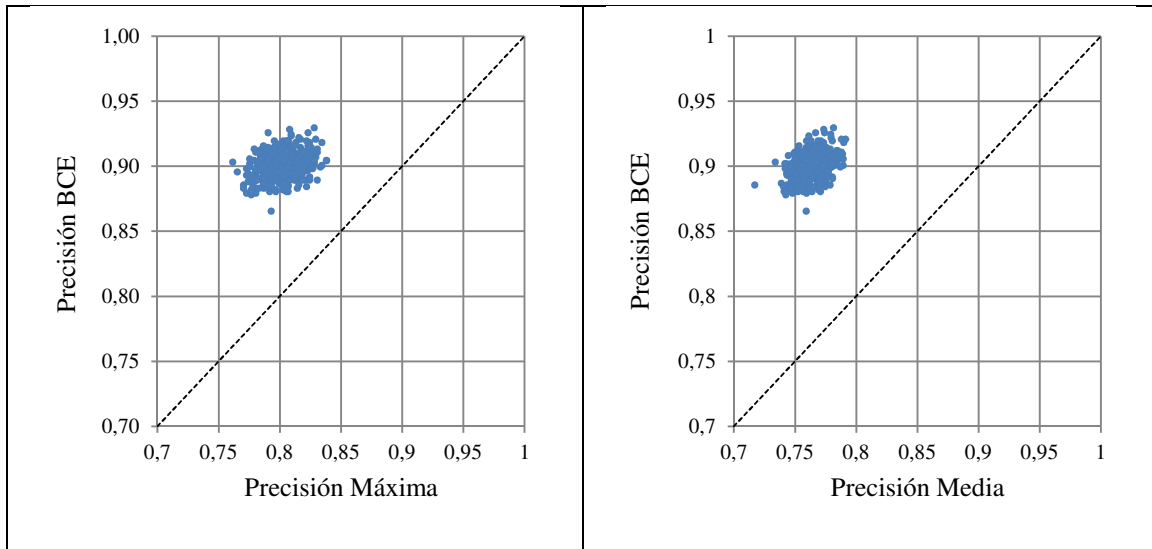


Figura 5.19. Semeion: Precisión Máxima y Precisión Media de los módulos base frente a la Precisión de CCE. La diagonal indica los casos en los que  $P_{CCE} = P_{MÁXIMA/MEDIA}$ .

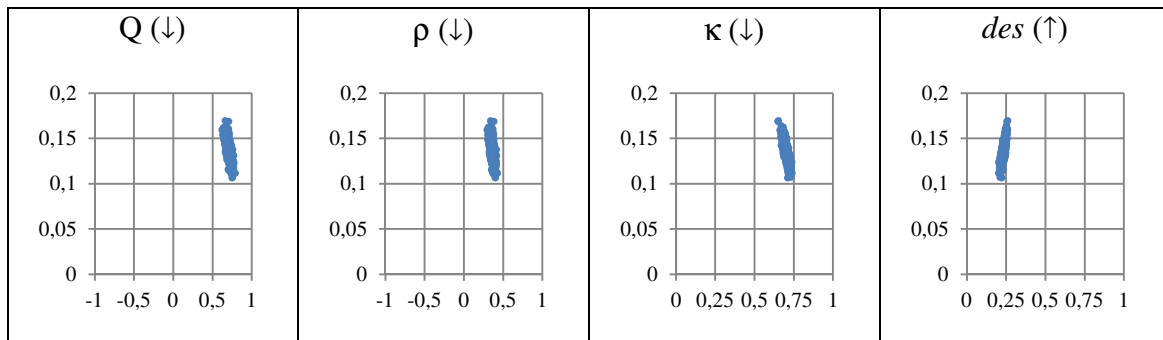


Figura 5.20. Mejora de CCE ( $P_{CCE} - P_{MEDIA}$ ) frente a 4 medidas de diversidad ( $Q$ ,  $\rho$ ,  $\kappa$ ,  $des$ ). El símbolo  $\uparrow/\downarrow$  indica que cuanto mayor es el valor de la medida mayor/menor es el grado de diversidad.

**Tabla 5.10. Coeficiente de Correlación de *Pearson* y Coeficiente de Correlación de *Spearman* entre la mejora del conjunto ( $P_{CJTO}-P_{MEDIA}$ ) y las distintas medidas de diversidad ( $Q$ ,  $\rho$ ,  $\kappa$ ,  $des$ ).**

	Coef. Corr. Pearson		Coef. Corr. Spearman	
	<i>CCE</i>	<i>BCE</i>	<i>CCE</i>	<i>BCE</i>
$Q$	-0,725	-0,491	-0,701	-0,456
$\rho$	-0,630	-0,442	-0,615	-0,404
$\kappa$	-0,820	-0,447	-0,814	-0,405
<i>des</i>	0,819	0,489	0,809	0,450

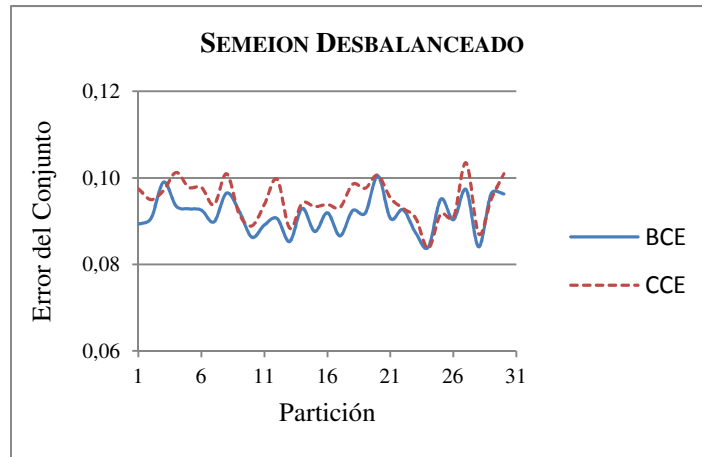
Por tanto, y aunque en términos de diversidad CCE parece mejor alternativa que BCE, los valores de precisión obtenidos indican que, al trabajar sobre la base de datos SEMEION y contrariamente a lo que se esperaba, BCE es ligeramente superior a CCE.

### 5.3. SEMEION MODIFICADO: Desbalanceo entre clases

Una vez analizado el comportamiento de BCE y CCE sobre una base de datos con un número relativamente alto de categorías, en este epígrafe se analiza el comportamiento de ambos modelos cuando el problema al que se enfrentan se caracteriza por un alto *desbalanceo* entre clases. Para realizar este estudio, se ha optado por modificar cuantitativamente los datos almacenados en la base de datos SEMEION y forzar a que, en este nuevo dominio, tres de las clases contengan una cuarta parte de los ejemplos originales (40 de 160). En concreto hemos optado por realizar el desbalanceo en las clases  $C_0$ ,  $C_7$  y  $C_9$  con el objetivo de que la descompensación en el número de ejemplos por clase afectara: i) a un clase fácil de aprender ( $C_0$ ) y ii) a dos clases,  $C_7$  y  $C_9$ , cuyos ejemplos tienden a ser confundidos con relativa frecuencia con instancias de otras clases (un "7" se confunde con relativa facilidad con un "1", y un "9" con un "4").

Para garantizar que la posible variación en los resultados solo esté condicionada por el desbalanceo inducido y no por otros factores, en este estudio se han mantenido las características específicas de BCE y CCE fijadas en el epígrafe 5.2. Así, todas las redes implementadas tendrán 20 neuronas en la capa oculta y se entrenarán durante 300 ciclos. Además, los clasificadores que componen los módulos base de BCE se integrarán siguiendo un esquema paralelo y CCE estará formado por 10 módulos base.

En la Figura 5.21 se muestra el porcentaje de error global obtenido por cada uno de los sistemas propuestos sobre cada una de las 30 iteraciones del proceso de validación cruzada con dos particiones.



**Figura 5.21. Porcentaje de error al evaluar BCE y CCE (10 módulos base) sobre la base de datos SEMEION DESBALANCEADA.**

Comparando las gráficas recogidas en la Figura 5.21 con las mostradas en la Figura 5.18 se puede comprobar que, a pesar del desbalanceo entre clases, el porcentaje de error del conjunto se mantiene prácticamente inalterable.

Para analizar el comportamiento de ambos sistemas sobre las clases minoritarias, en la Figura 5.22 se muestra el histograma respecto a la tasa de acierto [Ec. 5.4] de cada una de las clases minoritarias ( $C_0$ ,  $C_7$  y  $C_9$ ) y la tasa de acierto [Ec. 5.5] respecto al total de clases minoritarias:

$$Tasa\_acierto_i = \frac{TP_i}{N_i} \quad [Ec. 5.4]$$

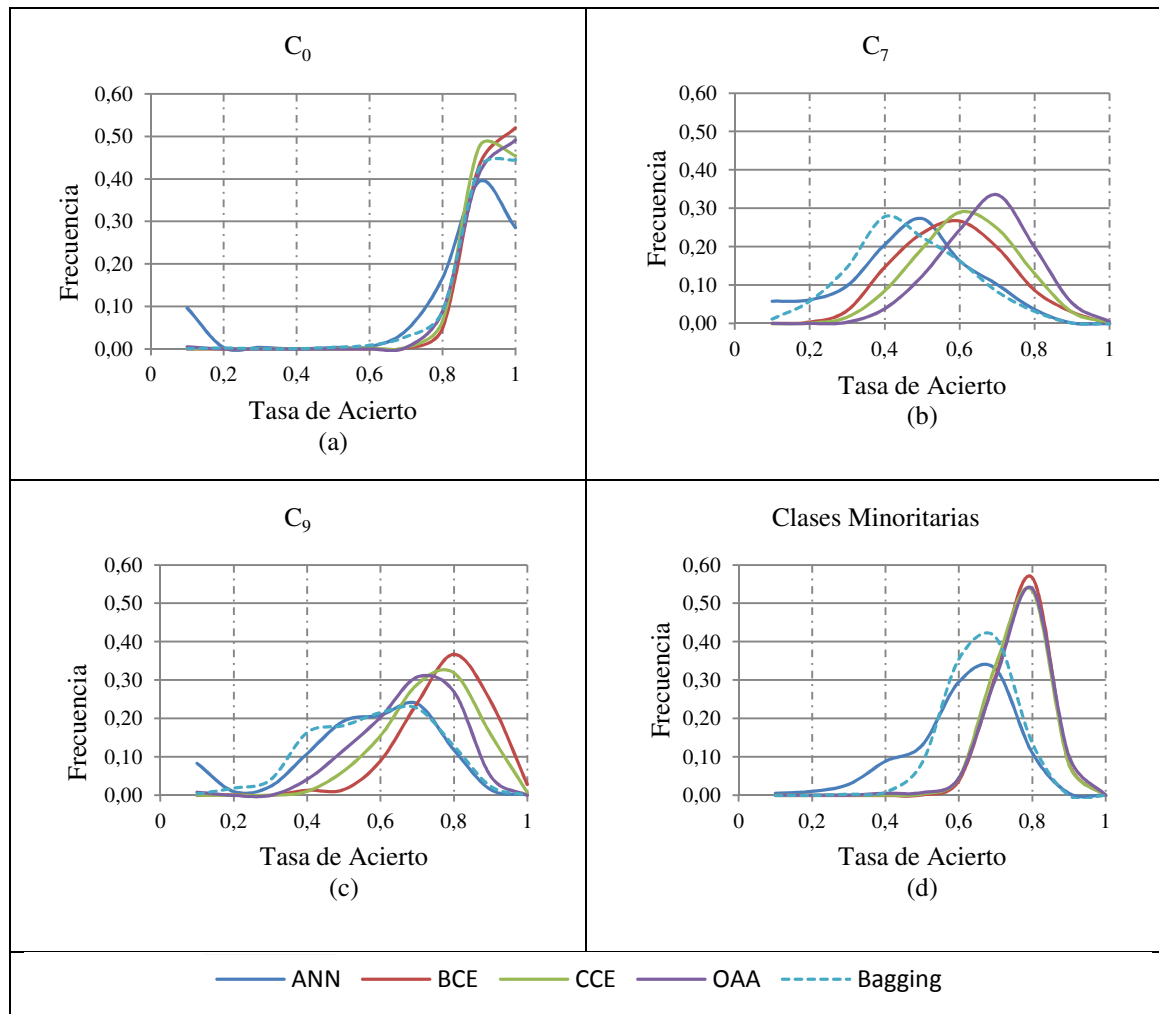
$$Tasa\_acierto_{GLOBAL} = \frac{\sum_i TP_i}{\sum_i N_i} \quad \forall i \in \text{clase minoritaria} \quad [Ec. 5.5]$$

donde:

$TP_i$ : Ejemplos pertenecientes a la  $i$ -ésima clase correctamente clasificados.

$N_i$ : Número de ejemplos pertenecientes a la  $i$ -ésima clase.

Junto a los valores obtenidos para BCE y CCE, y a título informativo, en la Figura 5.22 se muestran, también, los valores obtenidos al evaluar: una red de neuronas simple, la arquitectura OAA y *Bagging* (con 10 clasificadores base).



**Figura 5.22. SEMEION DESALANCEADO: Tasa de Acierto sobre las clases minoritarias.**

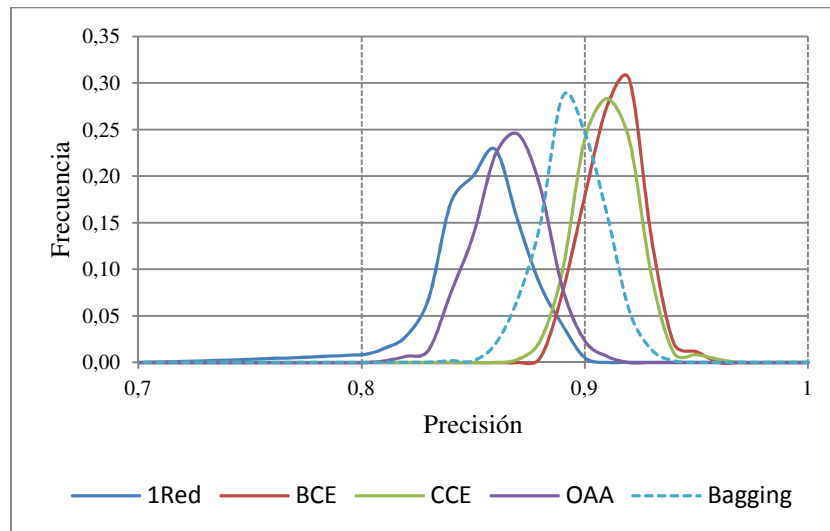
Analizando los valores recogidos en la Figura 5.22.(d) se puede observar que, al considerar la tasa de acierto sobre el total de las clases minoritarias, el comportamiento de BCE y CCE es prácticamente equivalente. Además, ambos comportamientos coinciden con el mostrado por la arquitectura OAA y mejoran los obtenidos con *Bagging* y con la red de neuronas simple. (Las curvas de BCE, CCE y OAA están desplazadas a la derecha lo que significa que la probabilidad de clasificar las clases minoritarias con una mayor tasa de acierto es más elevada.)

Al analizar el comportamiento por clases, se observa que:

- (a) Respecto a la clase  $C_0$  todos los sistemas tienen un comportamiento similar.
- (b) Respecto a la clase  $C_7$  el sistema que parece mostrar un mejor comportamiento es la arquitectura OAA seguida de CCE y BCE (las curvas están desplazadas hacia la derecha).
- (c) Respecto a la clase  $C_9$  el sistema que parece mostrar un mejor comportamiento es BCE seguido de CCE y *Bagging*.



Por último, y a título informativo, en la Figura 5.23 se muestra el histograma respecto a la Precisión (tasa de acierto considerando el total de clases), obtenido para todos y cada uno de los modelos de clasificación considerados.

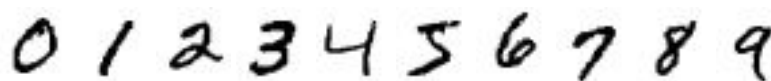


**Figura 5.23. SEMEION DESBALANCEADO: Histograma de la Precisión obtenida con distintos sistemas**

A la vista de estos resultados se puede concluir que, sobre el dominio considerado, BCE y CCE consiguen valores de precisión mejores que los que se obtienen con otros sistemas referenciados en la Bibliografía. Además, los resultados obtenidos al considerar únicamente las clases minoritarias parecen indicar que ambos sistemas son buenas alternativas sobre dominios en los que los ejemplos están altamente desbalanceados.

#### 5.4. MNIST: Diez clases y gran número de ejemplos de entrenamiento

Para evaluar la calidad de los sistemas propuestos cuando se enfrentan a un problema en el que la *cardinalidad* del conjunto de entrenamiento es elevada y bastante superior a la del conjunto de test, se ha usado la colección de datos MNIST [LeCun et al., 1995]. Esta base de datos contiene 60000 ejemplos de entrenamiento y 10000 ejemplos de test. Cada ejemplo se corresponde con un dígito manuscrito representado como una imagen de 28x28 píxeles en escala de grises (Figura 5.24). Se trata, por tanto, de un dominio en el que los ejemplos están inicialmente representados por 784 atributos y en el que cada ejemplo pertenece a 1 de 10 clases distintas (dígitos del 0 al 9).



**Figura 5.24. Ejemplos de la base de datos MNIST.**

Además, aunque el desbalanceo no es muy acusado (Tabla 5.11), en MNIST, la distribución de clases no es totalmente homogénea.

**Tabla 5.11. Distribución de ejemplos en la base de datos MNIST.**

Dígito	0	1	2	3	4	5	6	7	8	9
Entrenamiento	5923	6742	5958	6131	5842	5421	5918	6265	5851	5949
Test	980	1135	1032	1010	982	892	958	1028	974	1009
Total	6903	7877	6990	7141	6824	6313	6876	7293	6825	6958

Para facilitar la manipulación de los ejemplos y, en consecuencia, las fases de aprendizaje y test de los distintos clasificadores, cada instancia ha sido transformada de su formato original (estructura matricial en la que los enteros están almacenados en formato *MSB first -Most-Significant Byte first-*) a formato *PGM (Portable Gray Map)*.

A continuación, se detalla el procedimiento experimental seguido y se analizan los resultados obtenidos tras dicho proceso.

#### 5.4.1. Procedimiento experimental

Para analizar la viabilidad y la calidad de BCE y CCE, los resultados experimentales obtenidos con estas arquitecturas serán comparados con los obtenidos al aplicar los siguientes modelos de clasificación:

- Una única red de neuronas de una capa oculta y 10 nodos en la capa de salida.
- Un sistema de 10 redes de neuronas modelado según la arquitectura OAA.
- Bagging* [Breiman, 1996] con redes de neuronas como clasificadores base.
- ECOC* [Dietterich & Bakiri, 1995] implementado con redes de neuronas.

Dada la alta *cardinalidad* del conjunto de atributos y, en consecuencia, la alta probabilidad de que entre ellos exista cierta redundancia, cada uno de estos cuatro modelos ha sido construido y evaluado: i) con ejemplos descritos por todas sus características (784) y ii) con ejemplos sobre los que se ha realizado un proceso de selección de atributos. En este último caso, y con el objetivo de evitar sesgos atribuibles al método de selección de atributos, el algoritmo utilizado en este proceso será el usado en la construcción de CCE y BCE. Es decir, en todos los casos, el proceso de selección de características se ha realizado aplicando la versión de *CFS (Correlation-based Feature Selection)* [Hall, 1998], con *Best-First* [Russell & Norving, 2003] como algoritmo de búsqueda, implementada en la herramienta *WEKA*.

Para evaluar la precisión de los modelos implementados y determinar si las diferencias entre ellos son estadísticamente significativas, se ha optado por aplicar el *Test de McNemar* [Everitt, 1977]. Según se recoge en [Dietterich, 1997], este test constituye la mejor alternativa cuando el coste computacional de la experimentación impide realizar validación cruzada.

El *test de McNemar* permite comparar dos clasificadores,  $f_A$  y  $f_B$ , cuando éstos se entrenan y se testean sobre los mismos conjuntos de datos. Para aplicar este test es necesario construir la tabla de contingencia que aúna los errores/aciertos de  $f_A$  y los errores/aciertos de  $f_B$ . Es decir, construir la tabla esquematizada en la Figura 5.25.

$n_{00}$	$n_{01}$
$n_{10}$	$n_{11}$

Figura 5.25 Tabla de contingencia para el test de McNemar.

donde:

$n_{00}$ : Número de ejemplos incorrectamente clasificados por  $f_A$  y  $f_B$

$n_{01}$ : Número de ejemplos incorrectamente clasificados por  $f_A$  pero no por  $f_B$

$n_{10}$ : Número de ejemplos incorrectamente clasificados por  $f_B$  pero no por  $f_A$

$n_{11}$ : Número de ejemplos correctamente clasificados tanto por  $f_A$  como por  $f_B$

Bajo estas condiciones y aceptando que bajo la hipótesis nula los dos clasificadores tienen la misma tasa de error ( $n_{01}=n_{10}$ ), se puede demostrar que el estadístico:

$$\chi^2 = \frac{(|n_{01}-n_{10}|-1)^2}{n_{01}+n_{10}} \quad [\text{Ec. 5.6}]$$

sigue, aproximadamente una distribución  $\chi^2$  (*chi-cuadrado*) con un grado de libertad.

Por tanto, si el valor de  $\chi^2$  es mayor que el valor para el que la distribución  $\chi^2$  con un grado de libertad deja a su derecha un área  $\alpha$ , se podrá admitir, con un nivel de significancia  $\alpha$ , que los clasificadores comparados son estadísticamente equivalentes. En concreto, si  $\chi^2 \leq \chi^2_{1,0,05}=3,841$  se podrá asegurar con una confianza del 95% que los clasificadores analizados son estadísticamente equivalentes.

Por último, y para determinar si existe una relación entre la diversidad de los clasificadores base y la precisión de los distintos modelos, se calculará el valor de este parámetro aplicando algunas de las medidas referenciadas en la literatura. En concreto, junto a las medidas detalladas en la Tabla 5.4 (*estadístico Q*, *coeficiente de correlación*, *estadístico kappa*, y *medida de desacuerdo*), se calculará también, la *medida de fallo* y la *ambigüedad*. La definición y la relación teórica entre estas dos últimas medidas y la precisión del conjunto se resumen en la Tabla 5.12.

**Tabla 5.12. Definición de dos de las medidas de diversidad usadas en este problema. El símbolo  $\uparrow/\downarrow$  indica que cuanto mayor es el valor de la medida, mayor/menor es el grado de diversidad. La columna identificada por “D” indica si se trata de una medida dual (S) o no (N).**

Nombre	Símbolo	Definición	D	$\uparrow/\downarrow$
Medida de fallo	$f$	$\frac{1}{N} \sum_{n=1}^N \text{Diff}(C_i(\mathbf{x}_n), C_j(\mathbf{x}_n))$	S	$\uparrow$
Ambigüedad	$amb$	$\frac{1}{LNK} \sum_{l=1}^L \sum_{n=1}^N \sum_{k=1}^K \left( I_s(C_l(\mathbf{x}_k) = k) - \frac{N_k^n}{L} \right)^2$	N	$\uparrow$

donde:

$N$  es el número de ejemplos evaluados.

$L$  es el número de clasificadores base que integran el conjunto.

$K$  es el número de clases.

$C_i(x_n)$  es la clase que el clasificador  $i$  asigna al ejemplo  $x_n$ .

$N_k''$  es el número de clasificadores que indican que el ejemplo  $x_n$  pertenece a la clase  $k$  al ejemplo.

$Is()$  es un predicado de verdad.

En los siguientes apartados se describen en detalle las características de los modelos construidos, los resultados de su evaluación y la comparación estadística realizada.

#### 5.4.2. Características de los Modelos Evaluados

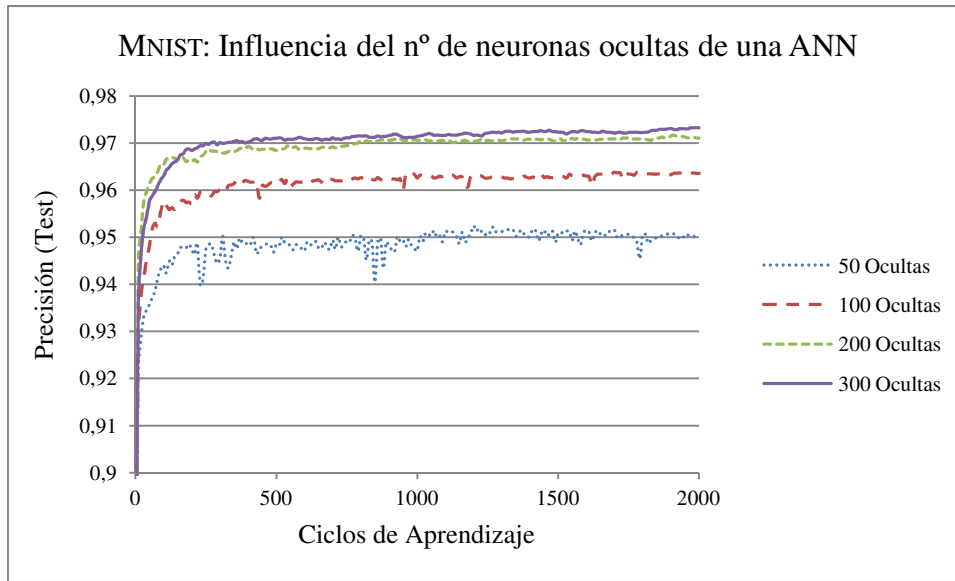
Para evaluar la calidad de los modelos propuestos cuando éstos se aplican sobre la base de datos MNIST, los resultados experimentales dados por BCE y CCE se compararán con los obtenidos al construir otros modelos de clasificación. En concreto, con los obtenidos al usar: una única red de neuronas, la *arquitectura OAA*, *Bagging* y *ECOC*.

En primera instancia, estos modelos de referencia se han construido partiendo de ejemplos sobre los que no se ha realizado ningún proceso de selección de características por lo que, en todos los casos, los clasificadores base tienen 784 neuronas en la capa de entrada. En una segunda fase, los sistemas propuestos han sido construidos partiendo de ejemplos descritos por un menor número de características. En este caso, y previo a la construcción de los distintos modelos de clasificación se ha analizado, para cada caso, el punto en el que se debe realizar este proceso. En otras palabras, se ha analizado si todos los clasificadores base han de construirse usando un único subconjunto de características o si, por el contrario, cada clasificador base ha de construirse usando un subconjunto de atributos específico (ver Figura 5.29).

A continuación se analizan los detalles específicos usados en la construcción de los clasificadores implementados.

#### Red de Neuronas Simple

Tras analizar para distintas arquitecturas el error cometido por la red de neuronas simple en función del número de ciclos de aprendizaje (Figura 5.26), y sopesar el coste computacional que lleva implícita cada una de las arquitecturas analizadas, se ha optado por trabajar con redes de 100 neuronas en la capa oculta y limitar el entrenamiento a 500 ciclos. Este número de ciclos se mantendrá cuando el aprendizaje se realiza a partir de ejemplos sobre los que se ha realizado un proceso de selección de atributos. No obstante, en este último caso, el número de neuronas en la capa oculta se reducirá a 50.



**Figura 5.26.** Precisión frente a Número de Ciclos de Aprendizaje para distintas arquitecturas de una Red de Neuronas evaluadas sobre el conjunto de test de MNIST.

Además, los pesos de las conexiones se han inicializado de forma aleatoria a valores comprendidos en el intervalo  $[-1, 1]$ , el algoritmo de aprendizaje es el *algoritmo de retropropagación*, el *coeficiente de aprendizaje* se ha fijado a 0,025 y la función de activación de las neuronas de la capa oculta y de la capa de salida es la función *sigmoideal*.

A menos que se señale lo contrario, la topología y arquitectura indicada en este apartado se aplicará a los clasificadores base usados en todos y cada uno de los conjuntos de clasificadores implementados para MNIST.

### Arquitectura OAA

Como ya se ha señalado en capítulos precedentes, la implementación de esta metodología pasa por construir tantos clasificadores binarios como clases existen. Cada uno de estos clasificadores se especializa en separar los ejemplos pertenecientes a una clase de los ejemplos pertenecientes a cualquiera de las restantes.

En el caso particular que nos ocupa, el sistema estará formado por 10 redes de neuronas con una única neurona en la capa de salida.

Dado que la salida dada por cada uno de los miembros del conjunto es un valor contenido en el intervalo  $[0, 1]$ , la salida global del sistema se obtiene seleccionando la clase asociado a la red que genere la salida de mayor valor (Ou & Murphey, 2007; Tax & Duin, 2002]. Matemáticamente,

$$c(\bar{x}) = F(\bar{x}, y_1, y_2, \dots, y_k) = \arg \max_{i=1, \dots, k} (y_i) \quad [\text{Ec. 5.7}]$$

donde  $y_i$  es la salida generada por el clasificador asociado a la  $i$ -ésima clase.

### **Bagging**

*Bagging* (**B**ootstrapping and **a**ggregating) [Breiman, 1996] es un conjunto de clasificadores en el que la diversidad de sus miembros se logra variando el conjunto de datos usado en la construcción de cada uno de ellos. En concreto, cada clasificador base se construye a partir de un subconjunto de ejemplos en el que, con respecto al conjunto de datos original, unas instancias están repetidas y otras omitidas. A pesar de su simplicidad y de su antigüedad (es uno de los primeros conjuntos de clasificadores propuestos en la literatura), se considera uno de los modelos de clasificación que ofrece mayor precisión [Polikar, 2006]. Además, su aplicación a problemas *multiclase* sólo está limitada por la naturaleza de sus clasificadores base.

Para determinar el número de clasificadores base de *Bagging* se ha intentado establecer un compromiso entre los 10 clasificadores propuestos por Quinlan [Quinlan, 1996], los 50 propuestos por Breiman [Breiman, 1996] y el coste computacional que implica la construcción de una red de neuronas. Basándonos en estos datos se ha optado por fijar el número de clasificadores en 20. Esta cantidad se ajusta bastante a la sugerida por [Opitz & Maclin, 1999] quienes concluyen que al implementar *Bagging* con Redes de Neuronas, la mayor reducción en el error ocurre al usar entre 10 y 15 clasificadores base.

Además, para evitar errores o aciertos que puedan atribuirse al método de integración [Duin & Tax, 2000], el método de combinación usado en *Bagging* coincidirá con el usado en BCE y CCE. Es decir, en los tres casos, la decisión final del sistema se obtiene (Ecuación. 5.8) promediando las salidas asociadas a cada clase y seleccionando la de mayor valor .

$$C(\bar{x}) = \arg \max_{i=1}^k \left( \frac{\sum_{j=1}^k y_{ji}}{k} \right) \quad [\text{Ec. 5.8}]$$

donde:

$y_{ji}$  es la salida asociada a la  $i$ -ésima clase dada por el  $j$ -ésimo clasificador.

$k$  es el número de clases.

### **ECOC**

ECOC (*Error Correcting Output Coding*) es un conjunto de clasificadores explícitamente diseñado para la resolución de problemas *multiclase* en el que la diversidad se consigue tras descomponer el problema original, de  $k$  categorías, en  $L$  *subproblemas* dicotómicos. En cada uno de estos *subproblemas* el conjunto de clases inicial,  $C=\{c_1, c_2, ..., c_k\}$ , se divide en dos subconjuntos,  $C_j^+$  y  $C_j^-$ , disjuntos. Los ejemplos asociados a las categorías contenidas en  $C_j^+$  son re-etiquetados con 1 y las instancias de cualquier clase contenida en  $C_j^-$  son re-etiquetadas con 0. Por tanto, el esquema de descomposición de clases usado por este método se puede representar mediante una matriz  $M$  de tamaño  $k \times L$  (Tabla 5.13) cuyos elementos toman valor  $\{0,1\}$  y en la que los ejemplos pertenecientes a la  $i$ -ésima clase serán ejemplos positivos para el  $j$ -ésimo clasificador si y solo si,  $m_{ij}=1$ .

En la fase de reconstrucción (determinación de la clase asociada a un ejemplo) y puesto que, en los casos que nos ocupan, los clasificadores base serán redes de neuronas cuya salida se puede interpretar como un estimador de la probabilidad de que el ejemplo pertenezca a cualquiera de las clases que son positivas para ese clasificador, la clase predicha por el sistema se determinará aplicando la ecuación 5.9:

$$C(\bar{x}) = \arg \min_{j=1, \dots, k} \sum_{i=1}^L |y_i - m_{i,j}| \quad [\text{Ec. 5.9}]$$

donde:

$y_i$  es la salida dada por el  $i$ -ésimo clasificador base.

$m_{i,j}$  es el  $j$ -ésimo elemento de la  $i$ -ésima columna de la matriz de códigos.

$L$  es el número de clasificadores base que integran la arquitectura.

$k$  es el número de clases.

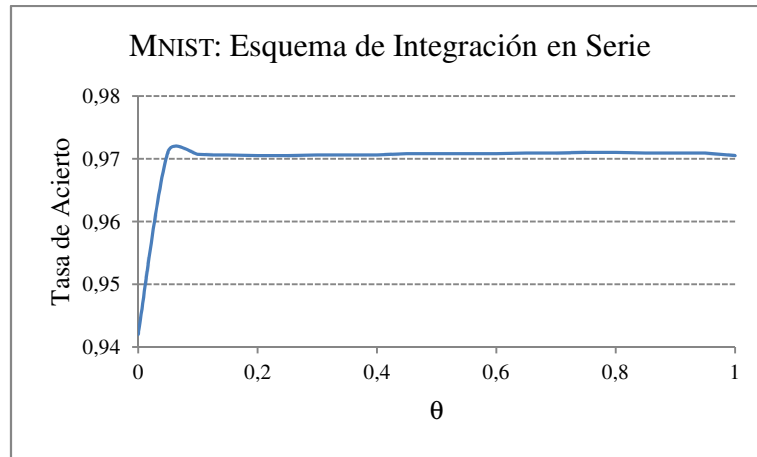
En la implementación de *ECOC* realizada para MNIST se ha usado el código de longitud 15 ( $L=15$ ) diseñado para un problema de 10 clases recogido en [Dietterich & Bakiri, 1995] y mostrado en la Tabla 5.13.

**Tabla 5.13. Código Corrector de Errores para el Reconocimiento de Dígitos Manuscritos.**

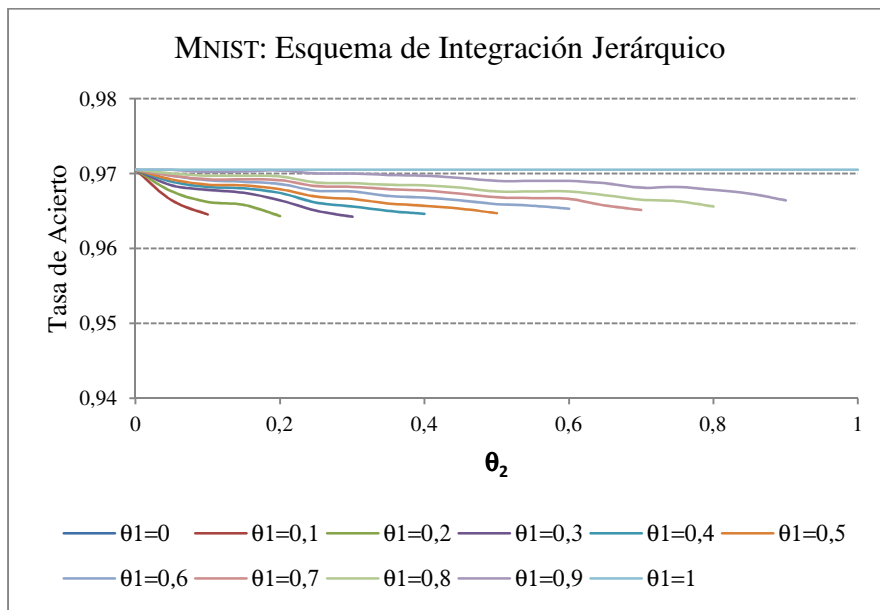
Clase	Código														
	f1	f2	f3	f4	f5	f6	f7	f8	f9	f10	f11	f12	f13	f14	f15
0	1	1	0	0	0	0	1	0	1	0	0	1	1	0	1
1	0	0	1	1	1	1	0	1	0	1	1	0	0	1	0
2	1	0	0	1	0	0	0	1	1	1	1	0	1	0	1
3	0	0	1	1	0	1	1	1	0	0	0	0	1	0	1
4	1	1	1	0	1	0	1	1	0	0	1	0	0	0	1
5	0	1	0	0	1	1	0	1	1	1	0	0	0	0	1
6	1	0	1	1	1	0	0	0	0	1	0	1	0	0	1
7	0	0	0	1	1	1	1	0	1	0	1	1	0	0	1
8	1	1	0	1	0	1	1	0	0	1	0	0	0	1	1
9	0	1	1	1	0	0	0	0	1	0	1	0	0	1	1

## BCE

Dado que la implementación de BCE exige conocer el esquema de integración Binario-Complementario usado en sus módulos base, a continuación y de forma grafica se muestran los resultados experimentales obtenidos al evaluar el esquema de integración en serie (Figura 5.27) y el esquema de integración jerárquico (Figura 5.28).



**Figura 5.27.** MNIST: Tasa de Acierto de BCE en función de  $\theta$  para el esquema de integración en Serie.



**Figura 5.28.** MNIST: Tasa de Acierto de BCE en función de  $\theta_1$  y  $\theta_2$  para el Esquema de Integración Jerárquico

Observando los valores representados en las Figuras 5.27 y 5.28 se puede comprobar que, de nuevo:

- Para valores de  $\theta \geq 0,1$  el esquema de integración en serie es invariable al valor de  $\theta$  y equivalente al esquema de integración en paralelo ( $\theta=0$ ).
- El esquema de integración jerárquico nunca es mejor que el esquema de integración en paralelo ( $\theta_1=1$  o  $\theta_2=0$ ).



Por tanto, y al igual que en los estudios precedentes, en lo que sigue y por simplicidad, BCE se implementará siguiendo un esquema de integración Binario-Complementario en paralelo.

## CCE

Puesto que el número de clases que definen MNIST es 10, por definición, el número de módulos que componen CCE debe ser un valor acotado en el intervalo  $[10, 20]$ . Con el fin de hacer comparable el coste computacional de CCE con el de las restantes arquitecturas, el número de módulos de este modelo se ha hecho coincidir con el número de clases. Es decir, se ha fijado en 10. Asimismo, la distribución de clases asociada a cada módulo se ha establecido de forma aleatoria garantizando que cada uno de los clasificadores base se construye considerando 5 de las 10 clases posibles. Además, para el caso particular que nos ocupa, al establecer la descomposición de clases de los distintos módulos se ha impuesto como condición que, para dos clases cualesquiera, existan, al menos, dos clasificadores base entrenados con ejemplos de ambas clases<sup>2</sup>.

Siguiendo la nomenclatura expuesta en el apartado 4.3.1, la matriz de descomposición de clases usada en la implementación de CCE es la recogida en la Tabla 5.14.

**Tabla 5.14. Matriz de descomposición de clases de CCE para el dominio MNIST. En esta codificación los ceros representan las clases usadas en uno de los clasificadores base del correspondiente módulo y los unos las clases usadas en el otro clasificador.**

	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9
M1	0	1	1	1	1	0	1	0	0	0
M2	0	1	1	1	0	1	1	0	0	0
M3	1	0	1	0	1	1	1	0	0	0
M4	1	0	1	0	0	0	1	1	0	1
M5	0	0	0	1	0	1	0	1	1	1
M6	1	1	0	1	1	0	0	0	1	0
M7	1	1	1	1	0	0	0	0	0	1
M8	1	1	0	0	1	1	0	0	0	1
M9	1	0	1	0	0	0	1	0	1	1
M10	0	0	1	0	0	1	0	1	1	1

Al igual que en BCE y *Bagging*, la salida global del conjunto se obtendrá usando la función promedio (Ec. 5.8)

<sup>2</sup> Esta condición es más restrictiva que la indicada en el epígrafe 4.3.2, donde se señala que al establecer la descomposición de clases, se ha de garantizar que, para dos clases cualesquiera, ha de existir, al menos, un clasificador base entrenado con ejemplos de ambas clases.

### 5.4.3. Resultados experimentales

Una vez descrito el método experimental y los modelos de aprendizaje implementados, en este apartado se muestran los resultados obtenidos en la evaluación de cada uno de ellos.

Como se señala en el apartado 5.4.1. para determinar si las diferencias observadas al evaluar dos modelos dados son o no significativas se usará el *Test de McNemar*. Dado que para aplicar este estadístico se requiere determinar el número de errores cometidos por un clasificador pero no por el otro, todos los resultados experimentales mostrados en este epígrafe se corresponderán con los obtenidos al usar una única versión de cada modelo. Es decir, en la evaluación experimental se obviará la influencia que sobre los resultados pueden tener los parámetros libres de cada modelo.

Para facilitar la comprensión del análisis, el estudio realizado se divide en dos partes. En la primera, se comparan los resultados de BCE y CCE con los obtenidos al construir una red de neuronas simple, la arquitectura OAA, *Bagging* y ECOC cuando estos últimos modelos se construyen usando ejemplos descritos por todas sus características. En la segunda fase se muestran los resultados obtenidos cuando todos los clasificadores base se construyen sobre ejemplos sobre los que se ha realizado un proceso de selección de atributos.

En la Tabla 5.15 se recogen los valores experimentales obtenidos en la primera fase de la experimentación.

**Tabla 5.15. Resultados experimentales obtenidos sobre MNIST por los distintos modelos de clasificación evaluados. En las diez primeras filas se muestra el porcentaje de ejemplos correctamente clasificados para cada una de las clases. En la última fila se muestra el porcentaje de ejemplos correctamente clasificados respecto al total de ejemplos. En negrita se indican los mejores valores.**

	1 ANN	OAA	Bagging	ECOC	BCE	CCE
C0	98,57%	<b>99,29%</b>	98,98%	98,88%	99,08%	<b>99,29%</b>
C1	98,94%	<b>99,03%</b>	98,85%	98,77%	98,85%	98,94%
C2	95,54%	96,71%	96,71%	97,09%	96,51%	<b>97,38%</b>
C3	96,34%	<b>97,82%</b>	97,13%	97,62%	97,13%	97,33%
C4	<b>98,17%</b>	97,86%	97,35%	97,86%	96,54%	96,84%
C5	95,52%	<b>96,86%</b>	95,85%	96,75%	95,85%	95,85%
C6	97,49%	97,60%	97,18%	<b>97,91%</b>	97,49%	97,70%
C7	95,33%	<b>96,98%</b>	96,40%	<b>96,98%</b>	96,60%	96,89%
C8	94,66%	<b>97,33%</b>	96,61%	<b>97,33%</b>	96,92%	96,61%
C9	93,86%	95,64%	95,64%	<b>96,33%</b>	95,24%	95,94%
Global	96,47%	97,53%	97,10%	<b>97,57%</b>	97,05%	97,31%

Analizando el valor de precisión de los distintos modelos se puede observar que los modelos más precisos son aquellos que resuelven el problema *multiclase* descomponiéndolo en subproblemas binarios. Es decir, *ECOC* y la arquitectura *OAA*.

En la Tabla 5.16, se recogen los resultados obtenidos al aplicar el *Test de McNemar* sobre los modelos evaluados. El resumen indicando las veces que, según los resultados

dados por este estadístico, un modelo gana/empata/pierde frente al resto de modelos se muestra en la Tabla 5.17.

**Tabla 5.16. MNIST: Errores cometidos por cada modelo y valores del test de McNemar.** Los valores de la diagonal principal (mostrados sobre fondo gris) indican el número de errores cometidos por cada modelo. Los valores de la matriz triangular superior indican el número de ejemplos mal clasificados por  $f_A$  –fila- y por  $f_B$  –columna- ( $n_{00}$ ). Los valores por debajo de la diagonal se corresponden con el valor obtenido al aplicar el test de McNemar (Ec. 5.6). El símbolo (+)/(-) indica que  $f_A$  es significativamente mejor/peor que  $f_B$ . El símbolo (.) indica que entre los modelos comparados no hay diferencias estadísticamente significativas.

	1 ANN	OAA	Bagging	ECOC	BCE	CCE
1 ANN	<b>353</b>	148	193	167	176	156
OAA	36,27 (+)	<b>247</b>	178	153	158	153
Bagging	14,96 (+)	9,75 (-)	<b>290</b>	190	209	185
ECOC	45,34 (+)	0,05 (.)	13,83 (+)	<b>243</b>	176	161
BCE	10,98 (+)	9,77 (-)	0,10 (.)	14,00 (-)	<b>295</b>	203
CCE	22,22 (+)	2,10 (.)	2,11 (.)	3,29 (.)	3,96 (+)	<b>269</b>

**Tabla 5.17. MNIST: Número de veces en las que, según el Test de McNemar, un sistema gana/empata/pierde frente a los demás.**

	Gana	Empata	Pierde
1 ANN	0	0	5
OAA	<b>3</b>	<b>2</b>	<b>0</b>
Bagging	1	2	2
ECOC	<b>3</b>	<b>2</b>	<b>0</b>
BCE	1	1	3
CCE	2	3	0

Observando los resultados recogidos en la Tabla 5.17 se puede concluir que, según el *Test de McNemar*, los mejores sistemas son OAA y ECOC seguidos de CCE. En un tercer grupo estarían *Bagging* y BCE, siendo el peor de todos los modelos analizados la red de neuronas simple.

Sin embargo, al analizar los tiempos de aprendizaje y test de cada uno de los modelos (Tabla 5.18), se observa que el tiempo de cómputo de BCE y CCE es muy inferior al del resto de modelos analizados. Por tanto, se puede concluir que, al combinar eficiencia y eficacia, BCE y CCE suponen una muy buena alternativa para la resolución del problema de aprendizaje presentado.

**Tabla 5.18. Tiempos de clasificación y aprendizaje de cada clasificador base medidos con el comando *time* de Linux en un Ordenador Intel(R) Core(TM) i7-2600 CPU @ 3.40GHz y detalles de la arquitectura de los métodos implementados. Para BCE y CCE se indican los valores para cada uno de los clasificadores que integran el primer módulo. Como número de entradas se refleja el valor medio.**

	Tiempo de clasificación (10000 eje.)	Tiempo de aprendizaje (500 ciclos)	N. de módulos	Neuronas entrada	Pesos (Capa 1)	Neuronas ocultas	Pesos (Capa 2)	Neur. de salida
1 ANN	0m2,210s	131m8,26s	1	784	78400	100	1000	10
OAA	0m8,450s	147m37,32s	10	784	78400	100	100	1
Bagging	0m17,070s	135m40,96s	20	784	78400	100	1000	10
ECOC	0m12,380s	142m12,69s	15	784	78400	100	100	1
BCE	0m5,290s	7m57,310s	10	83	4150	50	50	1
		17m13,700s	10	204	10200	50	450	9
CCE	0m3,320s	9m39,710s	10	160	8000	50	250	5
		6m17,610s	10	150	7500	50	250	5

Una vez analizado el comportamiento de los sistemas de referencia cuando éstos se construyen usando ejemplos descritos por todas sus características, a continuación se presenta el análisis realizado cuando dichos modelos se construyen usando ejemplos sobre los que se ha realizado un proceso de selección de atributos.

Dado que cada uno de los clasificadores base que integran *OAA*, *Bagging* y *ECOC* se construye a partir de un determinado subconjunto de ejemplos el primer aspecto a analizar es si el proceso de selección de características debe realizarse antes o después de generar estos subconjuntos. Como se puede comprobar en la Figura 5.29, si el proceso de selección de características se realiza antes de generar los subconjuntos usados en la construcción de cada clasificador base, todos y cada uno de estos clasificadores aprenderán de ejemplos descritos por el mismo subconjunto de características. Por el contrario, si el proceso de selección de atributos se realiza una vez obtenidos los distintos subconjuntos de entrenamiento, cada uno de los clasificadores base trabajará sobre un determinado subconjunto de características<sup>3</sup>.

En la Tabla 5.19 se presenta el error obtenido por cada modelo cuando éstos se construyen:

- usando un único subconjunto de características, y
- usando un conjunto de características específico para cada clasificador base

así como el valor del *Test de McNemar* obtenido al comparar ambas metodologías.

<sup>3</sup> Para el caso de BCE y CCE este estudio carece de sentido pues, por definición, y en el afán de fomentar la diversidad, todos y cada uno de los clasificadores que integran el conjunto han de construirse usando un subconjunto de atributos específico.

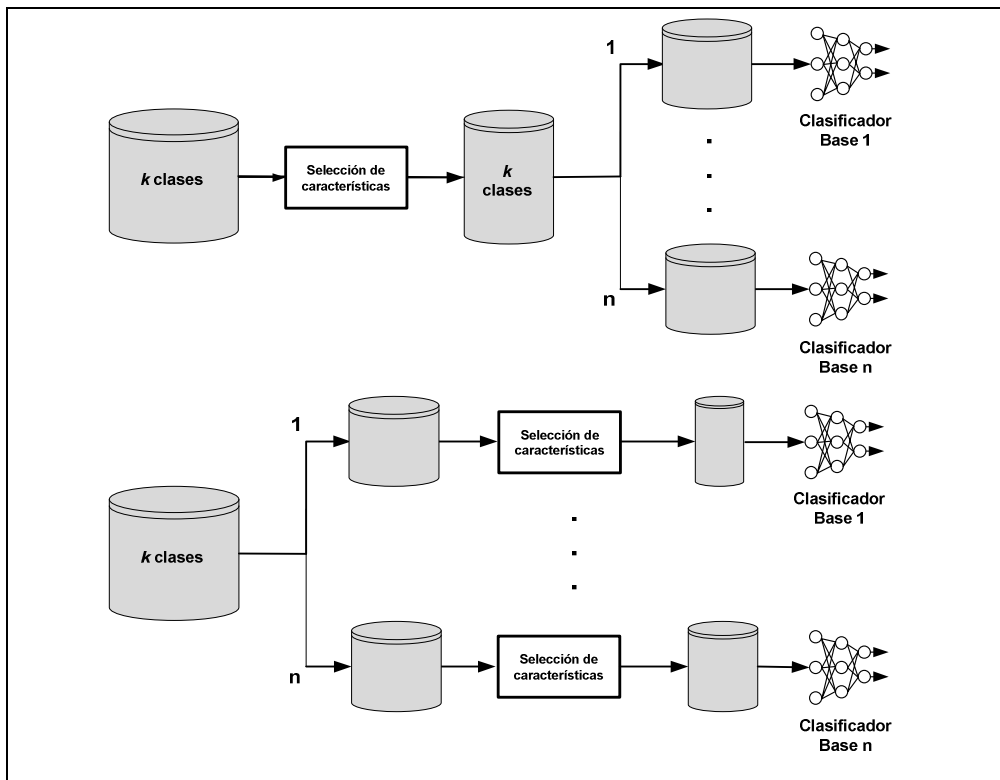


Figura 5.29. Esquemas de construcción de los clasificadores base cuando éstos se construyen usando un subconjunto de las características que describen los ejemplos. En el esquema de la parte superior, todos los clasificadores base se construyen usando un mismo subconjunto de características. En el esquema de la parte inferior, cada clasificador base usa un subconjunto de características específico.

Tabla 5.19. MNIST: Errores obtenidos al evaluar el método de selección de características sobre OAA, *Bagging* y ECOC.

	Selección única	Selección múltiple	Test de McNemar
OAA	<b>218</b>	568	110,557
<i>Bagging</i>	378	<b>333</b>	10,579
ECOC	<b>308</b>	474	72,020

Analizando los valores recogidos en la Tabla 5.19 se comprueba que cuando los clasificadores base se construyen usando un único subconjunto de características la precisión de OAA y ECOC mejora mientras que la de *Bagging* empeora. Por tanto, y dado que, según el *Test de McNemar*, las diferencias obtenidas son, en todos los casos, estadísticamente significativas, en lo que sigue, usaremos la versión de OAA y ECOC construida a partir de un único conjunto de características y la versión de *Bagging* en la que cada clasificador base se construye a partir de un subconjunto de atributos específico.

En la Tabla 5.20 se muestran los valores obtenidos al evaluar los distintos conjuntos de clasificadores cuando éstos se construyen a partir de ejemplos sobre los que se ha realizado un proceso de selección de atributos.

**Tabla 5.20. Resultados experimentales obtenidos sobre MNIST por los distintos modelos de clasificación cuando todos ellos se construyen a partir de ejemplos sobre los que se ha realizado un proceso de selección de atributos. En las diez primeras filas se muestra el porcentaje de ejemplos correctamente clasificados para cada una de las clases. En la última fila se muestra el porcentaje de ejemplos correctamente clasificados respecto al total de ejemplos. En negrita se indican los mejores valores.**

	1 ANN	OAA	Bagging	ECOC	BCE	CCE
C0	98,06%	98,88%	98,88%	98,98%	99,08%	<b>99,29%</b>
C1	98,15%	<b>99,12%</b>	98,85%	98,85%	98,85%	98,94%
C2	94,28%	96,32%	96,32%	95,93%	96,51%	<b>97,38%</b>
C3	94,85%	95,74%	96,63%	96,74%	97,13%	<b>97,33%</b>
C4	95,62%	<b>97,05%</b>	96,64%	96,84%	96,54%	96,84%
C5	93,83%	<b>96,41%</b>	94,84%	95,96%	95,85%	95,85%
C6	95,30%	97,18%	97,08%	97,44%	97,49%	<b>97,70%</b>
C7	95,23%	96,69%	96,01%	96,50%	96,60%	<b>96,89%</b>
C8	94,15%	95,48%	95,89%	96,51%	<b>96,92%</b>	96,61%
C9	92,47%	94,15%	95,14%	95,14%	95,24%	<b>95,94%</b>
Global	95,24%	96,73%	96,67%	96,92%	97,05%	<b>97,31%</b>

Analizando el valor de precisión obtenido para los distintos modelos se puede observar que, en este caso, los modelos más precisos son los propuestos en este trabajo de Tesis Doctoral, es decir CCE y BCE.

Para comprobar si las diferencias observadas son o no estadísticamente significativas, en la Tabla 5.21 se muestran los resultados obtenidos al aplicar el *Test de McNemar* sobre los modelos descritos. Para facilitar el análisis de estos resultados en la Tabla 5.22 se indican las veces que, según el estadístico aplicado, un sistema gana/empata/pierde respecto a los demás.

**Tabla 5.21. MNIST: Errores cometidos por cada modelo y valores del test de McNemar. Los valores de la diagonal principal (mostrados en negrita) indican el número de errores cometidos por cada modelo. Los valores de la matriz triangular superior indican el número de ejemplos mal clasificados por  $f_A$  –fila- y por  $f_B$  –columna- ( $n_{00}$ ). Los valores por debajo de la diagonal se corresponden con el valor obtenido al aplicar el test de McNemar (Ec. 5.6). El símbolo (+)/(-) indica que  $f_A$  es significativamente mejor/peor que  $f_B$ . El símbolo (.) indica que entre los modelos comparados no hay diferencias estadísticamente significativas.**

	1 ANN	OAA	Bagging	ECOC	BCE	CCE
1 ANN	<b>476</b>	208	246	243	221	198
OAA	56,60 (+)	<b>327</b>	206	181	190	181
Bagging	63,61 (+)	0,10 (.)	<b>333</b>	224	247	206
ECOC	76,61 (+)	1,34 (.)	3,21 (.)	<b>308</b>	195	177
BCE	98,49 (+)	3,98 (+)	10,21 (+)	0,78 (.)	<b>295</b>	203
CCE	121,60 (+)	13,88 (+)	20,89 (+)	6,53 (+)	3,96 (+)	<b>269</b>

**Tabla 5.22.** Número de veces en las que, según el Test de McNemar, un sistema gana/empata/pierde frente a los demás.

	Gana	Empata	Pierde
1 ANN	0	0	5
OAA	1	2	2
Bagging	1	2	2
ECOC	1	3	1
BCE	3	1	1
CCE	<b>5</b>	<b>0</b>	<b>0</b>

Observando los valores recogidos en las Tablas 5.21 y 5.22 se puede concluir que, cuando todos los sistemas se construyen a partir de ejemplos sobre los que se ha realizado un proceso de selección de características CCE es superior al resto de modelos considerados. Por su parte, BCE es peor que CCE y equivalente a ECOC. Por último indicar que, al contrario de lo que sucede al trabajar con ejemplos descritos por todas las características, las arquitecturas basadas en descomposición binaria (OAA y ECOC) parecen no ser una buena alternativa al problema planteado.

Para comprobar si existe una relación entre la diversidad de los clasificadores base y la precisión de los distintos modelos construidos, en la Tabla 5.23 se recogen los valores de precisión y diversidad obtenidos al evaluar *Bagging* (con y sin selección de características), BCE y CCE<sup>4</sup>.

**Tabla 5.23** Valores de diversidad y precisión para *Bagging*, BCE y CCE sobre MNIST. En negrita se indican los mejores valores.

<i>Modelo</i>	$f(\uparrow)$	$des(\uparrow)$	$Q(\downarrow)$	$\rho(\downarrow)$	$\kappa(\downarrow)$	$amb(\uparrow)$	<i>Precisión</i>
Bagging_784	0,0431	0,0355	0,9726	0,5636	0,9569	0,0041	97,10%
Bagging_BFCFS	0,0603	0,0499	0,9558	0,5184	0,9397	0,0057	96,67%
BCE	0,0821	0,0731	0,8926	0,3799	0,9179	0,0074	97,05%
CCE	<b>0,2246</b>	<b>0,2001</b>	<b>0,5630</b>	<b>0,2065</b>	<b>0,7742</b>	<b>0,0202</b>	<b>97,31%</b>

Analizando los valores mostrados en la Tabla 5.23 se puede comprobar que, en todos los casos, los valores de diversidad obtenidos están bastante alejados de los valores teóricos. No obstante, los sistemas con mayor valor de diversidad son BCE y, en especial, CCE.

<sup>4</sup> El cálculo de diversidad sólo se realiza sobre aquellos modelos en los que los clasificadores base son redundantes, es decir, sobre aquellos modelos en los que cada clasificador base aporta, por sí mismo, una solución no ambigua al problema de clasificación. Por tanto, junto a la red de neuronas simple (que no es estrictamente un conjunto de clasificadores), en el cálculo de la diversidad, se excluyen la arquitectura OAA y ECOC pues, en ambos casos, la salida dada por uno solo de sus clasificadores base no constituye una solución al problema de clasificación.

#### 5.4.4. Conclusiones

Los resultados obtenidos al analizar el comportamiento de BCE y CCE sobre MNIST indican que ambos modelos son una buena alternativa a la resolución de este problema.

Los datos mostrados en las Tablas 5.15, 5.16 y 5.17 reflejan que, en este dominio, los modelos que ofrecen mejores resultados, en lo que a precisión se refiere, son aquellos basados en descomposición binaria (OAA y ECOC). Sin embargo, los valores de tiempo recogidos en la Tabla 5.18 reflejan que la mejora que estos sistemas ofrecen frente a los propuestos en este trabajo se ve claramente descompensada por el coste computacional que implica su construcción. Por tanto, sopesando precisión y coste computacional se puede concluir que la mejor alternativa a este problema es la arquitectura CCE.

Los valores de diversidad recogidos en la Tabla 5.23 parecen demostrar que la calidad de los sistemas propuestos es consecuencia de la diversidad inducida en los módulos que los integran. En concreto, se puede observar que, de los sistemas comparados, el que mejor valor presenta tanto en diversidad como en precisión es CCE.

#### 5.5. REUTERS: 101 clases distribuidas jerárquicamente.

Para evaluar el comportamiento de las propuestas realizadas en esta Tesis, cuando se enfrentan a un problema con un gran número de clases organizadas jerárquicamente, se ha escogido como dominio la base de datos REUTERS CORPUS VOLUMEN 1 versión 2 (RCV1-v2) [Lewis et al., 2004]. Esta base de datos proviene de un archivo que contiene las 806791 noticias [Rose et al., 2002] publicadas por la Agencia Británica *Reuters Limited*<sup>5</sup> desde el 20 de Agosto de 1996 al 19 de Agosto de 1997. El archivo de noticias original fue filtrado, preprocesado y reorganizado por [Lewis et al., 2004] hasta ser transformado en una base de datos con 23149 ejemplos de entrenamiento (noticias publicadas entre el 20/08/1996 al 31/08/1996) y 781265 ejemplos de test (noticias publicadas entre el 01/09/1996 y el 19/08/1997). Todos los ejemplos de esta base de datos, están organizados conforme a tres esquemas de clasificación ortogonales: *Temática*, *Sector Empresarial* y *Zona Geográfica*. No obstante, en este trabajo, solo consideraremos el primero de estos esquemas pues es el único que tiene una estructura jerárquica y el utilizado en la mayor parte de las investigaciones que tratan con esta base de datos [Esuli, et al., 2008].

Atendiendo a la *Temática* de cada documento, existen 101 categorías distintas. Estas categorías están organizadas jerárquicamente, existiendo 4 categorías en el *Nivel 1* o nivel principal (CCAT, MCAT, ECAT y GCAT) y cuatro niveles de profundidad (*Nivel 1* a *Nivel 4*).

En las Figuras 5.30 y 5.31 se muestra el número de ejemplos de entrenamiento y test asignados a cada categoría así como el nivel jerárquico (representado por colores) al que pertenece cada una de ellas.

---

<sup>5</sup> <http://www.reuters.com>



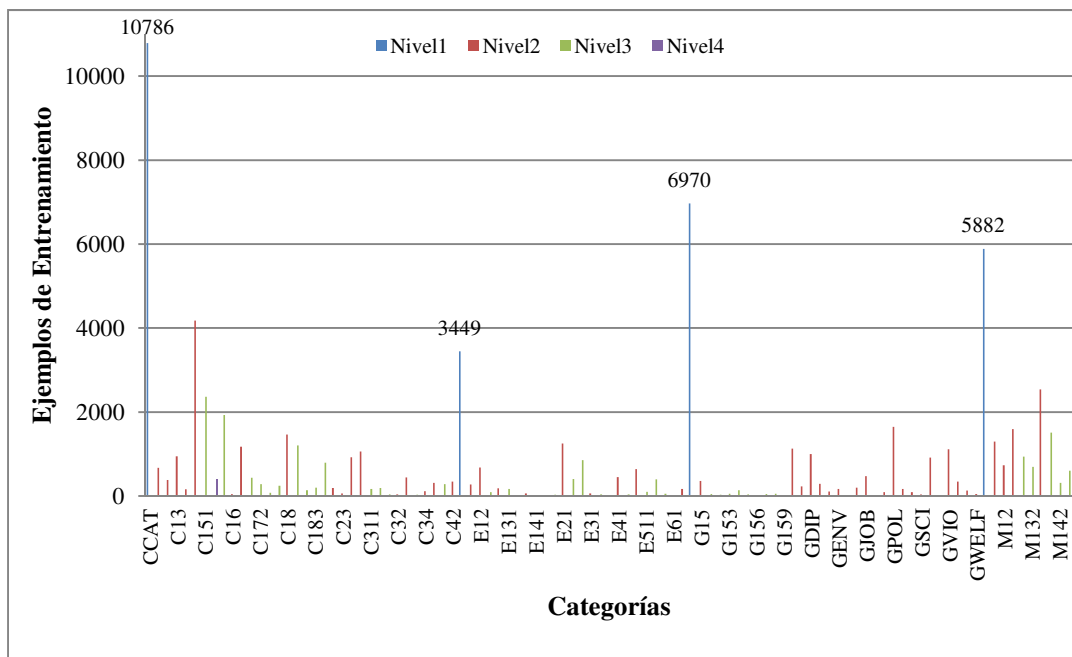


Figura 5.30. RCV1-v2- Ejemplos de entrenamiento asignados a cada categoría.

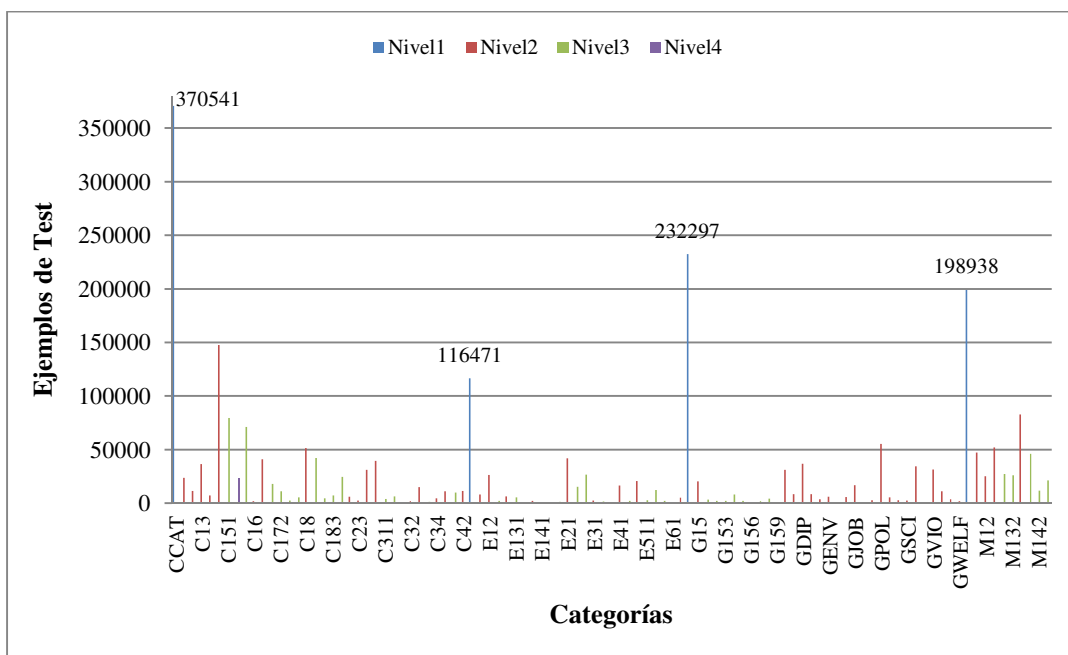


Figura 5.31. RCV1-v2- Ejemplos de test asignados a cada categoría.

Tras la fase de *preprocesamiento* realizada por los creadores de RCV1-v2, cada texto queda representado por el *peso* asignado a 47219 *palabras clave*. Es decir, cada ejemplo está representado por 47219 atributos continuos, si bien, en los ficheros suministrados en [Lewis, 2004] no se incluyen los atributos con valor 0.

Recapitulando lo indicado en este epígrafe, las principales características de la base de datos LYRL2004 (versión *preprocesada* de RCV1-v2) usada en esta fase experimental son:

1. El número de ejemplos de entrenamiento es 23149.
2. El número de ejemplos de test es 781265.
3. Los ejemplos están descritos por 47219 atributos con valor real y acotado en el intervalo  $[0; 1]$ .
4. Existen 101 categorías organizadas jerárquicamente en 4 niveles de profundidad.
5. Cada ejemplo puede pertenecer a una o varias categorías. En este último caso, las clases pueden estar incluidas dentro de una misma rama jerárquica, en ramas con un ascendiente común, o incluso pertenecer a ramas no relacionadas jerárquicamente. No obstante, dada la estructura jerárquica, se garantiza que si un ejemplo pertenece a una categoría de *Nivel  $n+1$*  dicho ejemplo también pertenece a la correspondiente categoría del nivel superior (*Nivel  $n$* ).
6. Existe una gran descompensación en el número de ejemplos que pertenecen a cada clase. Esta descompensación está presente no solo a nivel global (todas las clases) sino, también, a nivel jerárquico (clases que pertenecen a una misma rama jerárquica) e incluso a nivel de grupo (categorías pertenecientes a un mismo nivel jerárquico y con ascendiente común). Así, si consideramos los ejemplos de entrenamiento y dentro de ellos la categoría *CCAT* y todos sus descendientes, el número de ejemplos por categoría varía de los 10786 ejemplos asignados a *CCAT* a los 31 asignados a *C<sub>331</sub>*. Asimismo, si consideramos las categorías de Nivel 2 descendientes de *CCAT*, el número de instancias asociadas a cada clase varía de los 4179 ejemplos de *C<sub>15</sub>* a los 43 ejemplos de *C<sub>32</sub>*.

Estas características hacen que el método experimental usado en esta fase de experimentación sea muy distinto al expuesto en los epígrafes anteriores. A continuación, se describe en detalle este método.

### 5.5.1. Procedimiento experimental

La elevada *dimensionalidad* del problema planteado condiciona que, en este caso, para aplicar cualquier método de clasificación basado en *Redes de Neuronas*, resulte imprescindible realizar un proceso de selección de características previo. Además, el hecho de que un ejemplo pueda pertenecer simultáneamente a más de una clase, obliga a definir nuevas medidas de evaluación y a usar estadísticos específicos para los problemas *multietiqueta*. Seguidamente, se analizan estos aspectos en mayor detalle.

#### Selección de características

Reiterando lo ya indicado, el alto número de atributos con los que se describe cada instancia obliga a que, previo a la construcción de cualquier sistema de clasificación basado en Redes de Neuronas, sea imprescindible realizar un proceso de selección de características. Además, hay que señalar que, en este caso, la herramienta WEKA [Witten & Frank, 2005] no aporta las funcionalidades necesarias para realizar este

proceso ya que no soporta ficheros de un tamaño tan elevado como los que se manejan en este dominio (el fichero de entrenamiento supera los 9 GB). Por estos motivos, el primer aspecto a abordar es la elección e implementación de un método de selección de atributos adecuado a este problema. En esta línea y apoyándonos en experimentaciones previas [Lewis, et al., 2004; Esuli, et al., 2008] se han evaluado e implementado varios métodos de selección de atributos consistentes en ordenar las distintas características conforme al valor de una de las siguientes medidas:

- **Chi cuadrado,  $\chi^2$ :** El valor asignado a cada característica se obtiene calculado el valor del estadístico  $\chi^2$  (Ec. 5.10) de cada característica respecto a todas y cada una de las clases y seleccionando (Ec. 5.11) el mayor de ellos [Yang & Pedersen, 1997]. Así, si:

- $a$ : es el número de ejemplos que contienen la característica  $t$  y pertenecen a la clase  $c_i$ ,
- $b$ : es el número de ejemplos que contienen la característica  $t$  pero que no pertenecen a la clase  $c_i$ ,
- $c$ : es el número de ejemplos que pertenecen a la categoría  $c_i$  pero que no contienen la característica  $t$ ,
- $d$ : es el número de ejemplos que ni contienen la característica  $t$  ni pertenecen a la clase  $c_i$  y
- $n$ : es el número total de ejemplos usados,

el valor  $\chi^2$  para el atributo  $t$  respecto a la clase  $c_i$  viene dado por:

$$\chi^2(t, c_i) = \frac{n(ad-bc)^2}{(a+b)(a+c)(b+d)(c+d)} \quad [\text{Ec.5.10}]$$

Al considerar todas las categorías, el valor asignado a cada característica se puede obtener seleccionando el valor máximo. Es decir:

$$\chi_{\max}^2(t) = \max_i \chi^2(t, c_i) \quad [\text{Ec. 5.11}]$$

Una vez calculado el valor de  $\chi_{\max}^2$  para cada característica, el subconjunto de atributos seleccionado estará constituido por las  $M$  características con mayor valor.

- **Frecuencia,  $F$ :** El valor asignado a cada característica se obtiene contabilizando el número de ejemplos en los que dicha característica tiene un valor distinto de 0.

De nuevo, una vez calculado el valor de  $F$  para cada atributo se seleccionan las  $M$  características con mayor valor.

Una modificación de esta métrica consiste en considerar la *frecuencia* respecto a las distintas clases. Es decir, admitir que:

$$F(t, c_i) = a \quad [\text{Ec. 5.12}]$$

donde  $a$  mantiene la definición dada para el cálculo de  $\chi^2$ .

De este modo, el valor asignado a cada característica se puede calcular como:

$$F(t) = \max_i F(t, c_i) \quad [\text{Ec. 5.13}]$$

- **Ganancia en Información,  $IG$ :** En este caso, la métrica utilizada [Esuli et al., 2008] está dada por:

$$IG(t, c_i) = \sum_{c \in \{c_i, \bar{c}_i\}} \sum_{t \in \{t_k, \bar{t}_k\}} P(t, c) \log \frac{P(t, c)}{P(t)P(c)} \quad [\text{Ec. 5.14}]$$

donde:

$P(x)$  es la probabilidad de que se dé  $x$ , y

$P(t, c)$  es la probabilidad de que se den simultáneamente  $t$  y  $c$

Una vez calculado el valor de este parámetro para todas las características respecto a todas las clases, el valor asignado a cada característica se obtendrá aplicando la Ecuación 5.15:

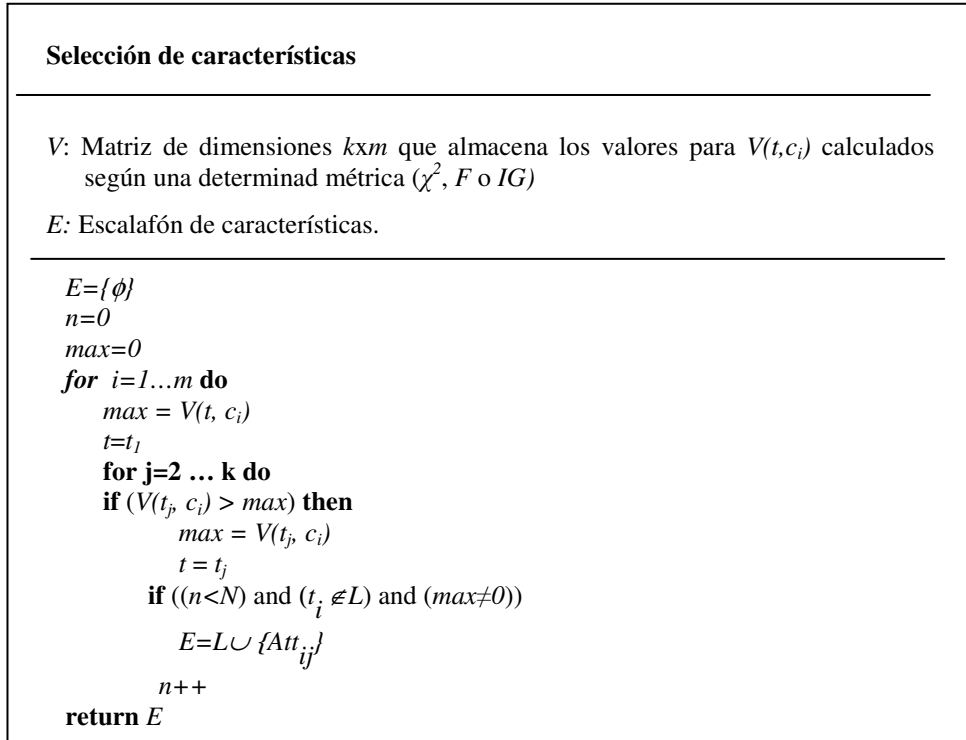
$$IG(t) = \max_i IG(t, c_i) \quad [\text{Ec. 5.15}]$$

Aunque los métodos anteriores se han aplicado con bastante éxito en problemas de clasificación de textos [Yang & Pedersen, 1995; Lewis, 2004; Castillo & Serrano, 2004] en [Forman, 2004] se señala que en todos los casos existe una problemática vinculada al método usado para asignar valores a cada atributo. Es decir, vinculada a las ecuaciones 5.9, 5.11 y 5.13. Según este autor, al aplicar las citadas ecuaciones, aquellos atributos que solo están presentes en las clases minoritarias y que, por tanto parecen tener gran relevancia frente a las mismas, tienen muy pocas probabilidades de ser seleccionados como atributos relevantes al conjunto. Así, este autor, propone que, una vez conocidos los valores  $V(t, c_i)$  ( $V$  es  $\chi^2$ ,  $F$  o  $IG$ ) para todos los atributos respecto a todas y cada una de las clases, se construya el conjunto de características relevantes seleccionando los  $v$  atributos que, respecto a cada categoría  $c_i$ , tienen el mayor valor  $V(t, c_i)$ . De este modo el conjunto de características seleccionado tendrá una *cardinalidad*  $q \leq vm$ , siendo  $m$  el número de categorías existente<sup>6</sup>.

Basándonos en el método de selección anteriormente descrito y con la pretensión de establecer, en función de cada una de las métricas, un *ranking* de características único, se ha implementado un algoritmo en el que se premia a aquellas características que mejor representan a cada clase. La Figura 5.32 muestra el algoritmo implementado.

---

<sup>6</sup> Dado que un mismo atributo puede ser relevante a más de una clase, en la mayoría de los casos  $q$  toma un valor menor que  $vm$



**Figura 5.32.** Algoritmo para establecer el *ranking* de atributos conforme al valor de  $V(t, c_i)$ .

Una vez determinada la métrica a utilizar ( $\chi^2$ ,  $F$  o  $IG$ ) y establecido el *ranking* de características, el subconjunto de atributos seleccionado incluirá aquellas características que ocupan los  $M$  primeros puestos del escalafón.

### Medidas de Evaluación

Hasta ahora, la calidad de los modelos implementados se ha establecido en función de la tasa de acierto o exactitud - en inglés, *accuracy*- del modelo (porcentaje de ejemplos correctamente clasificados frente al total de ejemplos). En dominios como el que nos ocupa, en los que los ejemplos pueden pertenecer de forma simultánea a más de un clase, las medidas más utilizadas para establecer la calidad de los modelos de clasificación son: la *precisión* ( $\pi$ ), la *sensibilidad* ( $\rho$ ) -en inglés, *recall*- y la *medida F* ( $F_1$ ) [Rijsbergen, 1979]. Estas tres medidas están definidas sobre cada categoría,  $c_i$ , y su cálculo matemático se realiza conforme a las ecuaciones 5.16, 5.17 y 5.18:

$$\pi_i = \frac{TP_i}{TP_i + FP_i} \quad [\text{Ec. 5.16}]$$

$$\rho_i = \frac{TP_i}{TP_i + FN_i} \quad [\text{Ec. 5.17}]$$

$$F_1^i = \frac{2TP_i}{2TP_i + FP_i + FN_i} = \frac{2\pi_i\rho_i}{\pi_i + \rho_i} \quad [\text{Ec. 5.18}]$$

donde:

$TP_i$ : Es el número de ejemplos que pertenecen a  $c_i$  correctamente clasificados por el modelo –*Verdaderos Positivos*–.

$FN_i$ : Es el número de ejemplos que pertenecen  $c_i$  pero que el sistema clasifica como pertenecientes a otra categoría –*Falsos Negativos*–.

$FP_i$ : Es el número de ejemplos que el modelo clasifica como pertenecientes a  $c_i$  pero que, realmente, pertenecen a otra clase distinta –*Falsos Positivos*–.

A la hora de establecer el rendimiento medio sobre todas las clases, el cálculo de estas medias se suele computar siguiendo uno de los siguientes esquemas [Yang, 1999]:

- *Macro-media*: En este caso, tras calcular los valores de  $\pi_i$ ,  $\rho_i$  y  $F_1^i$  para cada categoría, se obtiene el valor final (denotado por  $\pi^M$ ,  $\rho^M$  o  $F_1^M$ ) calculando el promedio sobre todas las clases (Ec. 5.19). Es decir, primero se evalúa localmente y luego globalmente.
- *Micro-media*: Para este esquema (denotado con el símbolo  $\mu$ ), el valor de  $\pi^\mu$ ,  $\rho^\mu$  o  $F_1^\mu$  sobre todas las clases se obtiene i) calculando el valor de  $TP_i$ ,  $FP_i$  y  $FN_i$  para cada una de las clases, ii) obteniendo el valor de  $TP$ ,  $FP$  y  $FN$  como la suma de todos los valores individuales y finalmente iii) aplicando la definición de la medida considerada (Ec. 5.20).

Así, particularizando para el caso de la medida  $F$ , las expresiones que nos permiten evaluar la calidad de un modelo son:

$$F_1^M = \frac{1}{k} \sum_{i=1}^k \frac{2TP_i}{2TP_i + FP_i + FN_i} \quad [\text{Ec. 5.19}]$$

$$F_1^\mu = \frac{2 \sum_{i=1}^k TP_i}{2 \sum_{i=1}^k TP_i + \sum_{i=1}^k FP_i + \sum_{i=1}^k FN_i} \quad [\text{Ec. 5.20}]$$

donde  $k$  indica el número total de clases.

## Comparación Estadística

Con el objetivo de determinar si las diferencias observadas al evaluar distintos modelos de clasificación sobre LYRL2004 son o no estadísticamente significativas, se usarán algunos de los estadísticos definidos por [Yang & Liu, 1999] para la clasificación de textos:

- **Micro sign test** (*s-test*): Compara dos clasificadores,  $A$  y  $B$ , analizando la clase asignada por cada uno de los sistemas a los distintos documentos que componen el conjunto de test.
- **Macro sign test** (*S-test*): Compara dos sistemas,  $A$  y  $B$ , analizando el valor de la medida  $F_1$  [Ec. 5.18] de cada uno de los sistemas sobre cada una de las categorías que integran el conjunto de test. Según este estadístico los sistemas serán equivalentes si  $F_1^A(c_i) > F_1^B(c_i)$  solo en la mitad de los casos en los que ambas medidas difieren. .
- **Macro T test** (*T-test*): Compara dos sistemas,  $A$  y  $B$ , contrastando el valor de la medida  $F_1$  [Ec. 5.18] de ambos sistemas para cada una de las categorías. Para

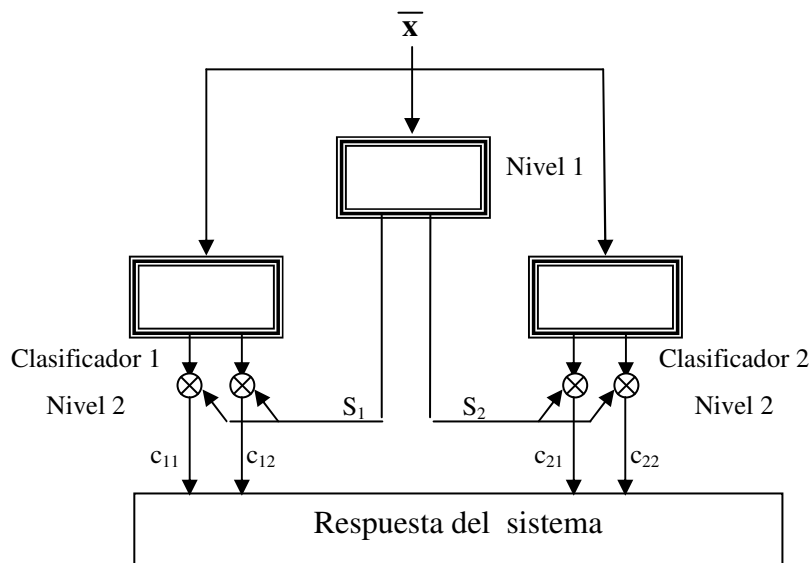
este estadístico los sistemas serán equivalentes si se puede demostrar que los valores de  $F_I$  de ambos sistemas representan poblaciones con igual valor medio.

- **Diferencia de proporciones ( $p$ -test):** Compara dos clasificadores,  $A$  y  $B$ , analizando los valores de  $\pi^u$  y  $\rho^u$  (Ecuaciones 5.15 y 5.16) de ambos sistemas.

La base teórica y la aplicación detallada de estos estadísticos se recogen en el Apéndice B de este trabajo.

### 5.5.2. Características de los Modelos Evaluados

Para evaluar la calidad de BCE y CCE cuando se enfrentan a la clasificación de textos, los resultados experimentales dados por este modelo se compararán con los obtenidos al aplicar otros modelos de referencia. Dado el carácter jerárquico del problema planteado, todos los modelos evaluados serán conjuntos de clasificadores en los que los clasificadores base se integran siguiendo una arquitectura jerárquica. Los detalles específicos de este esquema de integración se pueden consultar en el epígrafe 2.3. de este trabajo. No obstante, para facilitar la comprensión de este apartado, en la Figura 5.33, se muestra, nuevamente, una representación gráfica de este modelo de integración.



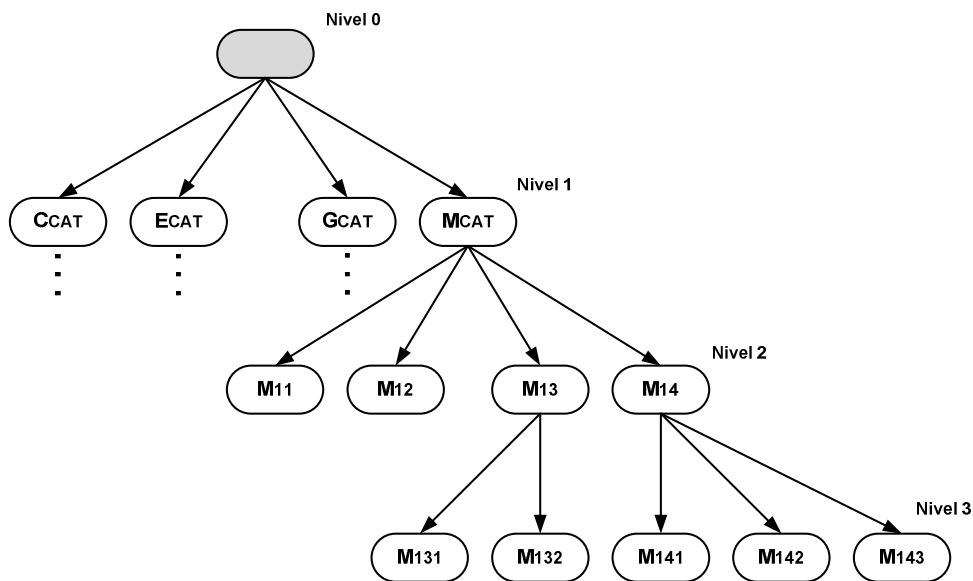
**Figura 5.33.** Conjunto de clasificadores integrados jerárquicamente usado en la resolución de un problema de 4 clases. El primer nivel está formado por un clasificador de 2 salidas y el segundo nivel por dos clasificadores con 2 salidas cada uno. La respuesta del sistema se obtiene multiplicando las salidas de ambos niveles. En este esquema el conjunto de clases se supone dividido en dos subconjuntos ( $S1=\{c11,c12\}$ ,  $S2=\{c21,c22\}$ ) cada uno de los cuales contiene 2 clases.

Así pues, todos los modelos de clasificación implementados para resolver el problema planteado en esta sección, se ajustarán al esquema recogido en la Figura 5.33. La diferencia entre unos modelos y otros estará en la metodología usada para construir los

distintos clasificadores base. A continuación se indican los detalles específicos de estos clasificadores para cada uno de los modelos implementados.

**Modelo1: ANN.** En este caso, cada uno de los clasificadores que integran la arquitectura será una *red de neuronas simple*. Cada una de estas redes se entrenará con los ejemplos pertenecientes a las clases que debe clasificar. Así, si consideramos únicamente la rama jerárquica que incluye a la clase *MCAT* y a todos sus descendientes (Figura 5.34), para poder clasificar todos los ejemplos incluidos en dicha rama se requerirá la construcción de las siguientes Redes de Neuronas:

- Una Red de Neuronas para el Nivel 1. Esta red se implementará a partir de los ejemplos pertenecientes a  $\{CCAT, ECAT, GCAT, MCAT\}$ .
- Una Red de Neuronas para el Nivel 2. Dicha red se construirá considerando únicamente los ejemplos pertenecientes a las clases  $\{M_{11}, M_{12}, M_{13}, M_{14}\}$ .
- Dos Redes de Neuronas para el Nivel 3. Cada una de estas redes se construirá a partir de las instancias asociadas a las clases  $\{M_{131}, M_{132}\}$  y  $\{M_{141}, M_{142}, M_{143}\}$ , respectivamente.



**Figura 5.34. LYRL2004: Representación gráfica de las categorías de la rama jerárquica que incluye a MCAT y a todos sus descendientes.**

Al considerar el total de clases (101) y la jerarquía existente entre ellas, este modelo estará constituido por 22 redes de neuronas (una red por cada subconjunto de clases que comparten progenitor).

Dado que, el conjunto de ejemplos, y por ende el conjunto de clases, usado en la construcción de cada una de las redes es distinto, el subconjunto de características más relevante a cada uno de ellos, obtenido según lo indicado en la sección 5.3.1, también



será distintito. Es decir, cada red se construirá usando un subconjunto de ejemplos y un subconjunto de atributos específico.

Como se acaba de indicar, en la construcción de cada red de neuronas solo se consideran los ejemplos asociados a las clases entre las que ha de discernir. Sin embargo, cuando en la estructura de clases existe un nodo que no tiene hermanos, la funcionalidad del clasificador consistirá en distinguir los ejemplos pertenecientes a la clase considerada de los ejemplos que, sin estar vinculados a ella, sí están asociados a su progenitor. Por tanto, y de forma excepcional, las redes de neuronas asociadas a este tipo de clases aprenderán de los ejemplos vinculados al progenitor. Si estos ejemplos, también, pertenecen a la categoría considerada se etiquetarán con un 1 y con un 0 en caso contrario.

En el proceso de selección de atributos, y tras analizar las métricas indicadas al comienzo de la sección 5.5.1, realizar algún experimento preliminar y consultar las recomendaciones dadas en la Bibliografía [Esuli, 2008], la *Ganancia en Información* ha sido el parámetro seleccionado para medir la relevancia de las distintas características.

En lo que respecta a la arquitectura de las *Redes de Neuronas* implementadas indicar que, en la fase preliminar, se analizaron distintas arquitecturas cada una de las cuales estaba caracterizada por el número de neuronas en la capa de entrada y el número de neuronas de la capa oculta. En concreto se implementaron cuatro configuraciones distintas: 2000×400, 1000×400 500×400 y 500×200. Analizando la calidad, en términos de  $\pi$ ,  $\rho$  y  $F_1$ , y el coste computacional de cada uno de estos modelos, se decidió seleccionar única y exclusivamente la última de estas configuraciones. Respecto a la topología, señalar que, tras una serie de experimentos preliminares, el valor del *coeficiente de aprendizaje* se ha fijado en 0,3 y el número de ciclos en los que se desarrolla el entrenamiento se ha fijado en 500 para la red del Nivel 1, 200 para las redes de Nivel 2 y 100 para las redes de Nivel 3.

Una vez construidas todas las redes que integran el modelo, y ya en la fase de clasificación, la salida asociada a cada una de las clases se obtiene multiplicando las salidas vinculadas a todos sus antecesores y analizando si dicho valor es mayor o menor que un umbral,  $\theta$ , previamente establecido. Así, tomando como ejemplo la clase  $M_{131}$  (ver Figura 5.34), la función que determina si un ejemplo dado pertenece o no a dicha clase será:

$$y_{M_{131}} = \begin{cases} 1 & \text{si, } S(M_{CAT}) * S(M_{13}) * S(M_{131}) > \theta \\ 0 & \text{en c. c.} \end{cases} \quad [\text{Ec. 5.21}]$$

donde, por convenio, el valor de  $\theta$  se ha fijado en 0,5 y:

$S(M_{CAT})$ : Es el valor de salida dado por la red de *Nivel1* asociado a  $M_{CAT}$ ,

$S(M_{13})$ : Es el valor de salida de la red de *Nivel2* asociado a  $M_{13}$  y

$S(M_{131})$ : Es el valor de salida de la red de *Nivel3* asociado a  $M_{131}$

Dado que esta filosofía no garantiza la clasificación de todos los ejemplos<sup>7</sup>, se impondrá como condición que, si tras calcular el valor de todas las salidas asociadas a un ejemplo ninguna de ellas tiene un valor superior a 0,5, el ejemplo será catalogado como perteneciente a la clase vinculada a la salida con mayor valor.

Algunas de las características dadas para la construcción de este primer modelo han sido usadas en el resto de modelos implementados por lo que, a continuación, solo se indican las características específicas del resto de arquitecturas implementadas.

**Modelo2: OAA.** En este caso, cada una de las redes de neuronas implementadas para el *Modelo1* ha sido sustituida por un conjunto de redes binarias que se ajusta al esquema *uno contra todos*. Así, por ejemplo, la red de cuatro salidas ({CCAT, ECAT, GCAT, MCAT}) construida para el *Nivel 1* del modelo anterior, se sustituye, en este modelo, por cuatro Redes de Neuronas de una única salida.

A la hora de establecer la codificación de clases de los ejemplos que han de servir para la construcción de cada una de las redes que integran este modelo se han asumido los siguientes supuestos:

1. En la construcción de cada red solo intervienen los ejemplos que comparten progenitor. Es decir, los ejemplos pertenecientes a una determinada clase y los asociados a las clases *hermanas*.
2. Aquellos ejemplos pertenecientes a la clase considerada serán etiquetados con un 1.
3. Los ejemplos pertenecientes a cualquiera de las clases hermanas se etiquetan con un 0.
4. Cuando la red está asociada a una clase que carece de hermanos, el conjunto de ejemplos de entrenamiento estará formado por los ejemplos vinculados al progenitor. Si estos ejemplos, pertenecen a la categoría considerada se etiquetan con un 1 y con un 0 en caso contrario.

**Modelo3: BCE.** En la construcción de este modelo, cada una de las redes de neuronas implementadas para el *Modelo1* ha sido sustituida por un conjunto de clasificadores que sigue el esquema BCE. Así, la red de cuatro salidas ({CCAT, ECAT, GCAT, MCAT}) construida para el *Nivel1* del *Modelo1*, se sustituye por 8 redes de neuronas cuya codificación de clases está esquematizada en la Tabla 5.24.

---

<sup>7</sup> No hay nada que garantice que, al menos, el valor de salida asociado a una de las clases sea superior a 0,5.

**Tabla 5.24.** Codificación de clases asociada a las redes que integran la arquitectura BCE usada en la construcción del clasificador de Nivel 1.

Binario	Complementario
CCAT, <b>NOT</b> (CCAT)	{ECAT, MCAT, GCAT}
ECAT, <b>NOT</b> (ECAT)	{CCAT, MCAT, GCAT}
MCAT, <b>NOT</b> (MCAT)	{CCAT, ECAT, GCAT}
GCAT, <b>NOT</b> (GCAT)	{CCAT, ECAT, MCAT}

Para el caso particular en el que el número de clases incluidas en un mismo grupo jerárquico es igual o inferior a dos, y dado que en estos casos la construcción de los clasificadores complementarios carece de sentido, se aplicarán las siguientes restricciones de diseño:

- Cuando una clase carece de hermanos, la tarea de clasificación vinculada a los ejemplos pertenecientes a esta categoría recaerá única y exclusivamente en el correspondiente clasificador binario.
- Cuando una clase solo tiene un hermano, la combinación Binario-Complementario asociada a ambas clases se limitará a dos clasificadores binarios construidos siguiendo las pautas especificadas para el caso de una clase que carece de hermanos. Así, si consideramos las clases  $M_{131}$  y  $M_{132}$  recogidas en la Figura 5.34 el esquema de clasificación encargado de discernir si un ejemplo pertenece a una de estas dos clases estará constituido por dos redes binarias. La primera de estas redes se construirá considerando como positivos los ejemplos que pertenecen a  $M_{131}$  y como negativos los ejemplos que pertenecen a  $M_{13}$  pero no a  $M_{131}$ . Asimismo, la segunda de estas redes se construirá considerando como positivos los ejemplos asociados a  $M_{132}$  y como negativos los ejemplos vinculados a  $M_{13}$  pero no a  $M_{132}$ .

**Modelo4: CCE.** En la construcción de este modelo, cada una de las redes de neuronas implementadas para el *Modelo1* se sustituye por un conjunto de clasificadores que sigue el esquema CCE. Así, el Nivel 1 estará integrado por 3 módulos base que se ajustan a la matriz de descomposición de clases indicada en la Tabla 5.25.

**Tabla 5.25. REUTERS: Matriz de descomposición de códigos para los módulos de CCE de Nivel 1**

	C <sub>CAT</sub>	E <sub>CAT</sub>	M <sub>CAT</sub>	G <sub>CAT</sub>
M <sub>1</sub>	0	0	1	1
M <sub>2</sub>	0	1	0	1
M <sub>3</sub>	0	1	1	0

Puesto que, la filosofía CCE sólo es aplicable en problemas (o subproblemas) en los que el número de clases es superior a dos, cuando el número de clases de un subnivel sea

inferior o igual a este valor, se aplicarán las restricciones de diseño expuestas, para estos casos, sobre BCE.

Uno de los problemas encontrados a la hora de evaluar este modelo sobre el conjunto de test es determinar el umbral,  $\theta$ , que permite decidir si un ejemplo pertenece o no a una clase dada [Ec. 5.21]. Experimentalmente se ha comprobado que, al fijar este umbral en 0,5 (valor por defecto), los valores obtenidos para  $\pi$  y  $F_1$  (especialmente a nivel *micro*) son muy inferiores a los logrados con el resto de modelos evaluados. Como ejemplo, en la Tabla 5.26 se muestran los valores de *precisión*, *sensibilidad* y de la *medida F* obtenidos al evaluar BCE y CCE sobre las clases de *Nivel I*<sup>8</sup>.

**Tabla 5.26. Valores de *precisión* ( $\pi$ ), *sensibilidad* ( $\rho$ ) y *Medida F* ( $F_1$ ) obtenidos al evaluar BCE y CCE sobre las clases Nivel I.**

	BCE			CCE		
	$\pi$	$\rho$	$F_1$	$\pi$	$\rho$	$F_1$
CCAT	0,939	0,916	0,927	0,584	0,985	0,733
ECAT	0,844	0,776	0,8089	0,371	0,912	0,527
GCAT	0,915	0,917	0,9163	0,594	0,970	0,737
MCAT	0,927	0,909	0,9183	0,625	0,968	0,760

Al intentar establecer el umbral de CCE a partir de los ejemplos de entrenamiento, los resultados sobre el conjunto de test empeoran. La única opción probada que supone una mejora de estos valores es fijar dicho umbral en función de un determinado subconjunto de ejemplos de test<sup>9</sup>. No obstante, y dado que esta opción implica un sesgo tanto en la evaluación final como en la subsecuente comparación estadística, la única opción posible es fijar este umbral mediante un proceso de validación cruzada en el que sólo intervengan los ejemplos de entrenamiento. Sin embargo, puesto que fijar este umbral para CCE y no hacerlo para el resto de modelos también supondría introducir un sesgo<sup>10</sup> en la evaluación experimental, se ha optado por excluir CCE del estudio que se describe en el siguiente epígrafe.

### 5.5.3. Resultados Experimentales

Una vez descritos los modelos de clasificación implementados para este problema, en esta sección se muestran los resultados obtenidos en la evaluación de cada uno de ellos. Para facilitar el estudio, en esta sección, únicamente se muestra el resumen de los resultados obtenidos y los resultados referidos a la comparación estadística realizada. Además, para dar mayor valor a estos resultados, en esta comparación, se incluyen los resultados suministrados por Lewis en [Lewis, 2004b] y que corresponden a la aplicación de *máquinas de soporte vectorial* (SVM) [Joachims, 1998], el algoritmo *Rocchio* [Rocchio, 1971] y el algoritmo del *vecino más cercano* (*k-NN*) [Martin, 1995]

<sup>8</sup> Los valores obtenidos sobre todas las clases están recogidos en la Tabla C.4 del Apéndice C.

<sup>9</sup> Se han considerado los 9966 primeros ejemplos del conjunto de test.

<sup>10</sup> Experimentalmente se ha probado que al fijar este umbral para BCE considerando los primeros 9966 ejemplos de test hay una variación en los resultados.

sobre la base de datos LYRL2004. La información detallada de los valores mostrados en este epígrafe se recoge en el Apéndice C.

La Tabla 5.27 muestra los valores de precisión ( $\pi$ ), sensibilidad ( $\rho$ ), y medida F ( $F_1$ ) calculadas sobre las tablas de contingencia (Apéndice C) obtenidas tras la aplicación de los distintos modelos evaluados.

**Tabla 5.27. Valores de precisión ( $\pi$ ), sensibilidad ( $\rho$ ), y medida F a nivel *micro* ( $\mu$ ) y *macro* (M) para los distintos modelos evaluados. En negrita se indican los mejores valores. En fondo gris se recogen los valores obtenidos con la arquitectura BCE.**

	$\pi^\mu$	$\rho^\mu$	$F_1^\mu$	$\pi^M$	$\rho^M$	$F_1^M$
ANN	0,805	0,770	0,787	0,639	0,515	0,550
OAA	0,802	0,768	0,785	0,668	0,502	0,544
BCE	0,833	0,777	0,804	0,680	0,521	0,566
knn_ $\mu$	0,781	0,761	0,771	0,615	0,524	0,516
knn_M	0,735	0,797	0,765	0,544	0,606	0,560
Rocchio	0,700	0,687	0,693	0,615	0,468	0,504
svm1_ $\mu$	<b>0,844</b>	0,790	<b>0,816</b>	<b>0,693</b>	0,561	0,579
svm1_M	0,829	<b>0,803</b>	<b>0,816</b>	0,656	<b>0,610</b>	<b>0,619</b>

Observando los valores recogidos en la Tabla 5.27 se puede comprobar que el modelo que ofrece mejores resultados es la *máquina de soporte vectorial* (*svm1*) descrita y evaluada en [Lewis et al., 2004]. No obstante, al excluir este modelo se puede inducir que el modelo que, en media, ofrece mejores resultados es aquel construido usando la arquitectura BCE (Modelo3).

Para verificar si estas observaciones son ciertas desde un punto de vista estadístico, a continuación (Tablas 5.28 a 5.32) se muestran los resultados obtenidos al aplicar los test estadísticos recomendados en los problemas de clasificación de textos (ver sección 5.5.1). Para facilitar la comprensión e interpretación de los resultados, adjunto a cada comparación se indica la conclusión extraída siguiendo la siguiente nomenclatura:

- (++) indica que  $C_A$  es mejor que  $C_B$  con un nivel de confianza del 99%.
- (--) indica que  $C_A$  es peor que  $C_B$  con un nivel de confianza del 99%.
- (+) o (-) indica que las diferencias son significativas a un nivel de confianza del 95%.
- (.) indica que, a un nivel de confianza del 95%, las diferencias observadas no son estadísticamente significativas.

Por último señalar que en el estudio estadístico realizado, BCE se compara con el resto de modelos indicados al comienzo de este epígrafe, excepto al aplicar el estadístico *s-test*. Para poder aplicar este estadístico se requiere conocer, de forma explícita, la clase o clases asignadas por cada clasificador a cada ejemplo. Puesto que esta información no se incluye en la información suministrada por Lewis, la comparación de sus modelos con BCE mediante el estadístico *s-test* resulta inviable.

En las siguientes tablas (Tabla 5.28 a Tabla 5.32) se muestran los valores intermedios y la conclusión final obtenidos al aplicar los distintos estadísticos.

**Tabla 5.28. Resultados obtenidos al aplicar el estadístico *s-test*.**

$C_A$	$C_B$	n	k	Z	Conclusión
BCE	ANN	395072	244262	148,679	(++)
BCE	OAA	623643	366633	138,814	(++)

**Tabla 5.29. Resultados obtenidos al aplicar el estadístico *S-test*.**

$C_A$	$C_B$	n	k	Z	Conclusión
BCE	ANN	95	78	6,258	(++)
BCE	OAA	94	74	5,570	(++)
BCE	knn_μ	98	69	4,041	(++)
BCE	knn_M	98	63	2,828	(++)
BCE	Rocchio	98	82	6,667	(++)
BCE	svm1_μ	97	26	4,975	(--)
BCE	svm1_M	95	85	7,695	(--)

**Tabla 5.30. Resultados obtenidos al aplicar el estadístico *T-test*.**

$C_A$	$C_B$	n	$\bar{d}$	$s_{\bar{d}}$	t	$t_{0,05}$	$t_{0,01}$	Conclusión
BCE	ANN	95	0,017	0,005	3,720	1,661	2,367	(++)
BCE	OAA	94	0,023	0,006	4,096	1,661	2,367	(++)
BCE	knn_μ	98	0,051	0,012	4,091	1,661	2,365	(++)
BCE	knn_M	98	0,006	0,008	0,777	1,661	2,365	(.)
BCE	Rocchio	98	0,062	0,010	6,087	1,661	2,365	(++)
BCE	svm1_μ	97	0,012	0,013	0,915	1,661	2,366	(.)
BCE	svm1_M	95	0,053	0,008	6,653	1,661	2,367	(--)

**Tabla 5.31. Resultados obtenidos al aplicar el estadístico *p-test* sobre los valores de *precisión* ( $\pi$ ).**

$C_A$	$C_B$	$\pi_A$	$\pi_B$	$n_A$	$n_B$	p	Z	Conclusión
BCE	ANN	0,8333	0,805	2364553	2423166	0,819	80,405	(++)
BCE	OAA	0,8333	0,802	2364553	2426801	0,818	88,671	(++)
BCE	knn_μ	0,8333	0,781	2364553	2469190	0,808	145,526	(++)
BCE	knn_M	0,8333	0,735	2364553	2745869	0,788	267,685	(++)
BCE	Rocchio	0,8333	0,7	2364553	2484933	0,768	346,056	(++)
BCE	svm1_μ	0,8333	0,844	2364553	2369045	0,839	-31,643	(--)
BCE	svm1_M	0,8333	0,829	2364553	2454993	0,831	12,596	(++)

**Tabla 5.32. Resultados obtenidos al aplicar el estadístico  $p$ -test sobre los valores de sensibilidad ( $\rho$ ).**

$C_A$	$C_B$	$\rho_A$	$\rho_B$	$n_A$	$n_B$	$p$	$Z$	Conclusión
BCE	ANN	0,777	0,77	2533121	2533121	0,774	18,821	(++)
BCE	OAA	0,777	0,768	2533121	2533121	0,773	24,161	(++)
BCE	knn_μ	0,777	0,761	2533121	2533121	0,769	42,723	(++)
BCE	knn_M	0,777	0,797	2533121	2533121	0,787	-54,975	(--)
BCE	Rocchio	0,777	0,687	2533121	2533121	0,732	228,682	(++)
BCE	svm1_μ	0,777	0,79	2533121	2533121	0,784	-35,523	(--)
BCE	svm1_M	0,777	0,803	2533121	2533121	0,790	-71,839	(--)

A continuación, en la Tabla 5.33 se resumen las conclusiones extraídas al aplicar los distintos estadísticos.

**Tabla 5.33. Conclusiones extraídas tras la aplicación de los distintos estadísticos. En fondo gris se indican los casos en los que la arquitectura BCE (Modelo3) es estadísticamente mejor que el modelo con el que se compara.**

$C_A$	$C_B$	s-test	S-test	T-test	p-test( $\pi$ )	p-test( $\rho$ )
BCE	ANN	(++)	(++)	(++)	(++)	(++)
BCE	OAA	(++)	(++)	(++)	(++)	(++)
BCE	knn_μ	?	(++)	(++)	(++)	(++)
BCE	knn_M	?	(++)	(.)	(++)	--
BCE	Rocchio	?	(++)	(--)	(++)	(++)
BCE	svm1_μ	?	(--)	(.)	(--)	(--)
BCE	svm1_M	?	(--)	(--)	(++)	(--)

Observando los valores recogidos en la Tabla 5.33 se puede comprobar que si bien la máquina de soporte vectorial desarrollada por Lewis [Lewis et al., 2004] es estadísticamente mejor que el modelo basado en la arquitectura BCE, este último modelo es superior al resto de modelos considerados.

#### 5.5.4. Conclusiones

El estudio experimental realizado sobre la base de datos LYRL2004 muestra que, al intentar resolver el problema de Clasificación de Textos usando clasificadores basados en Redes de Neuronas, los mejores resultados se alcanzan cuando los clasificadores base se ajustan a la arquitectura BCE. Además, se ha comprobado que los resultados ofrecidos por este modelo son estadísticamente mejores que los dados por algunos de los sistemas típicamente usados en la clasificación de textos. En concreto, y con un nivel de confianza del 99% se puede asegurar que, según los resultados dados por los estadísticos  $S$ -test y  $p$ -test evaluado sobre *precisión*, BCE es mejor que los modelos implementados por Lewis basados en el algoritmo *Rocchio* y en el algoritmo del *vecino más cercano* ( $k$ -NN). Al considerar el estadístico  $p$ -test evaluado sobre *sensibilidad*, sólo el algoritmo  $k$ -NN optimizado para *macro-medias* supera a BCE. Por último, al

considerar el estadístico *T-test*, se concluye que BCE es superior a *Rocchio* y a *k-NN* optimizado para *micro-medias* y equivalente a *k-NN* optimizado para *macro-medias*.

No obstante, los resultados empíricos también señalan que BCE es estadísticamente peor que el modelo basado en *máquinas de soporte vectorial* modelado por Lewis. Esta última conclusión hace plantearnos, como trabajo futuro, la necesidad de determinar la pertenencia o no de un ejemplo a una clase utilizando mecanismos más robustos (por ejemplo, el propuesto en [Addis, et al., 2008]) que los usados en este trabajo [Ec. 5.21].

## 5.6. ASISTENTUR: Un problema real.

El interés en este dominio está en la necesidad de construir sistemas de ayuda a la conducción que, entre otros factores, alerten al conductor de posibles peligros, impidan o impongan la realización de ciertas maniobras o delimiten la velocidad a la que se debe circular. La necesidad de integrar el módulo de clasificación en un sistema de ayuda a la conducción capaz de trabajar en tiempo real impone ciertas restricciones de diseño que dificultan su construcción. En concreto, y con el objetivo de que el sistema software diseñado pueda implementarse en hardware sobre una FPGA (*Field Programmable Gate Array*) se nos ha impuesto como condición que el algoritmo de clasificación esté basado en redes de neuronas y que el número de nodos de cada red no exceda de 200.

El dominio usado contiene 900 señales de prohibición distribuidas equitativamente en 9 categorías distintas (Figura 5.35): Prohibido el paso a peatones, prohibido girar a la izquierda/derecha, prohibido parar o estacionar, prohibido adelantar y prohibido circular a más de 60-50-40-20 y 100 km por hora.

Dado que las señales de tráfico proceden de fotogramas extraídos de un video grabado en movimiento en un entorno urbano, muchas de estas imágenes sufren de colusiones, sombras, y/o falta de nitidez que dificultan tanto la tarea de aprendizaje como la de clasificación. Además, cada una de las imágenes tiene unas dimensiones determinadas por lo que, previo a la construcción de cualquier modelo de aprendizaje, ha sido necesario realizar un *preprocesamiento* de las mismas. Tras dicho *preprocesamiento*, cada señal ha quedado representada por una imagen de 32x32 píxeles en formato PGM. De este modo se consigue que, desde el punto de vista cualitativo, la representación de los ejemplos sea similar a la usada en MNIST.



Figura 5.35. Ejemplo de los nueve tipos de señales de tráfico.



### 5.6.1. Procedimiento experimental

Por analogía con el procedimiento experimental descrito en el epígrafe 5.4.1. y con el fin de analizar la viabilidad y la calidad de las arquitecturas propuestas en esta Tesis, los resultados experimentales obtenidos por BCE y CCE serán comparados con los obtenidos al aplicar los siguientes conjuntos de clasificadores<sup>11</sup>:

- a) Una única red de neuronas de una capa oculta y 9 neuronas de salida.
- b) La arquitectura OAA.
- c) *Bagging*.
- d) *ECOC*.

Cada uno de estos cuatro modelos, se ha implementado usando redes de neuronas de una sola capa oculta con *Back-Propagation* como algoritmo de aprendizaje. Además, cada uno de ellos ha sido construido y evaluado con ejemplos descritos por todas sus características (1024) y con ejemplos sobre los que se ha realizado un proceso de selección de atributos mediante la aplicación del algoritmo CFS+BF.

Para evaluar la precisión de los modelos implementados y determinar si las diferencias entre ellos son estadísticamente significativas, se ha aplicado el estadístico  $F$  sobre los resultados obtenidos al ejecutar 5 réplicas del proceso de validación cruzada estratificada con 2 particiones [Alpaydin, 1999]. En cada réplica del proceso de validación cruzada, el conjunto de ejemplos se ha dividido aleatoriamente en dos subconjuntos de igual tamaño. Posteriormente, cada algoritmo de aprendizaje se ha entrenado sobre uno de estos subconjuntos de datos y se ha testeado sobre el otro. Bajo estas condiciones y asumiendo que  $p_i^{(j)}$  es la diferencia entre el porcentaje de error de dos de los clasificadores construidos usando el subconjunto  $j$  de la  $i$ -ésima réplica y  $s_i^2 = (p_i^{(1)} - \bar{p}_i)^2 + (p_i^{(2)} - \bar{p}_i)^2$  es la varianza estimada en dicha réplica ( $\bar{p}_i = (p_i^{(1)} + p_i^{(2)}) / 2$ ), entonces:

$$F - test = \frac{\sum_{i=1}^5 \sum_{j=1}^2 (p_i^{(j)})^2}{2 \sum_{i=1}^5 s_i^2} \quad [\text{Ec. 5.22}]$$

sigue una distribución  $F$  de 10 y 5 grados de libertad. Por tanto, si el valor de  $F-test$  es menor que  $F_{0.05}(10,5)=4.735$  se podrá asegurar con un grado de confianza del 95% que los modelos comparados son estadísticamente equivalentes. Por el contrario, si el valor de  $F-test$  es superior a 4.735 se podrá concluir que las diferencias observadas entre los modelos comparados son estadísticamente significativas.

Por último, y para determinar si existe una relación entre la diversidad de los clasificadores base y la precisión de los distintos modelos, se calculará el valor de este parámetro aplicando las métricas recogidas en las Tablas 5.4 y 5.11.

<sup>11</sup> Nuevamente, y para simplificar la terminología, la red de neuronas simple será considerada como un conjunto de clasificadores

En los siguientes apartados se describen en detalle las características de los modelos contruidos, los resultados de su evaluación y el análisis de la comparación estadística realizada.

### 5.6.2. Características de los Modelos Evaluados

Dada la analogía que a nivel cualitativo presentan los ejemplos incluidos en este dominio respecto a los de MNIST, los modelos implementados serán bastante similares a los descritos en el epígrafe 5.4.2. No obstante, las diferencias cuantitativas que hay entre ambas bases de datos obligan a introducir las modificaciones que se detallan a continuación.

#### Red de Neuronas Simple

En este caso, dadas las características del dominio se usará un *perceptrón* multicapa con 9 neuronas en la capa de salida y una única capa oculta. Tras realizar el correspondiente estudio preliminar el número de neuronas en la capa oculta se ha fijado en 50 cuando la red trabaja con ejemplos descritos por todas sus características y en 30 cuando la red trabaja con ejemplos sobre los que se ha realizado un proceso de selección de atributos. Por su parte, y por analogía con los estudios previos, los pesos de las conexiones se han inicializado de forma aleatoria a valores comprendidos en el intervalo  $[-1, 1]$ , el *coeficiente de aprendizaje* se ha fijado a 0,025 y la función de activación de las neuronas de la capa oculta y de la capa de salida es la función *sigmoideal*. El número de ciclos en los que se desarrolla el aprendizaje se ha fijado en 2000.

A menos que se especifique lo contrario, los parámetros indicados en este apartado (a excepción del número de neuronas en la capa de salida) son los usados en la construcción de los clasificadores base que integran el resto de conjuntos implementados.

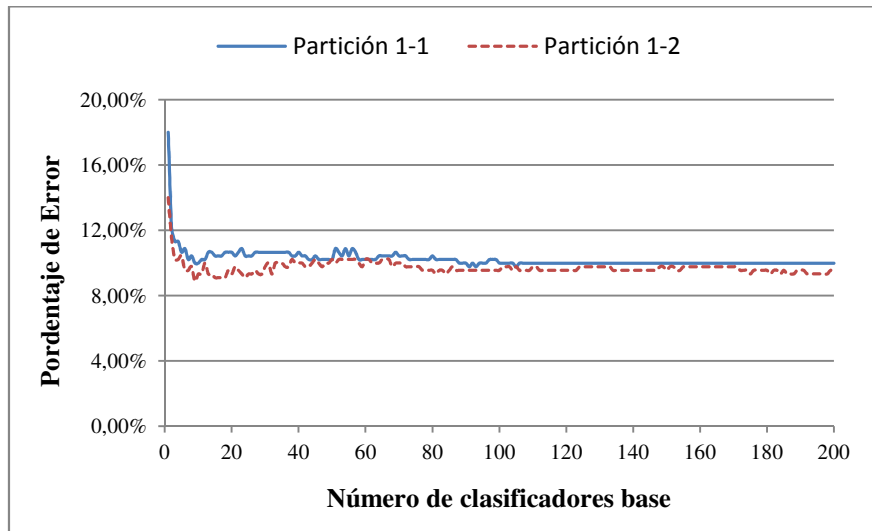
#### Arquitectura OAA

El esquema seguido en la implementación de esta arquitectura es análogo al indicado en el epígrafe 5.4.2. con la salvedad de que, en este caso, el número de clasificadores base será 9. Además, y al igual que para MNIST, y como se justifica más adelante, todos y cada uno de los clasificadores binarios se han construido partiendo del mismo subconjunto de atributos. En concreto, el subconjunto de atributos usado es el obtenido al considerar todos los ejemplos de entrenamiento.

#### Bagging

Como se recoge en el epígrafe 5.4.2, según los estudios realizados por [Optiz & Maclin, 1999], al implementar *Bagging* con redes de neuronas, la mayor reducción en el error ocurre al usar entre 10 y 15 clasificadores base. Al analizar el comportamiento de este sistema sobre este dominio se ha verificado que, en efecto, se satisface este comportamiento. No obstante, con el fin de garantizar la estabilidad del error, el número

de clasificadores base se ha fijado en 20. Además, y al igual que en MNIST, el método de integración usado para conocer la salida final del sistema es el promedio [Ec. 5.8].



**Figura 5.36. ASISTENTUR: Error en función del número de clasificadores de *Bagging* para alguna de las evaluaciones realizadas.**

## ECOC

Dado que los códigos *ECOC* suministrados por Dietterich<sup>12</sup> no contemplan el caso de un problema de 9 clases, para aplicar esta metodología, se ha optado por crear un *código corrector de error* (Tabla 5.34) siguiendo las pautas indicadas en [Dietterich & Bakiri, 1995]:

- Para aumentar la tolerancia a fallos el *número de clasificadores* base debe ser *superior al número de clases*. Puesto que en el caso que se presenta el número de clases es 9, el número de módulo se ha fijado en 15.
- Para garantizar una baja correlación entre los errores cometidos por los distintos clasificadores se requiere *maximizar la distancia Hamming* (número de bits en que difieren) *entre columnas y columnas complementarias*. En el código generado la distancia entre columnas varía entre 2 y 6.
- Para garantizar que la decisión generada por ECOC sea correcta aunque la clasificación emitida por  $(d-1)/2$  de los clasificadores sea errónea se requiere una *separación entre filas* (en términos de la distancia *Hamming*) *igual o superior a d*. En el código propuesto la distancia mínima entre filas es igual a 5 por lo que se garantiza que el sistema puede corregir hasta 5 errores individuales.

En la fase de reconstrucción, la clase asociada a cada ejemplo se calcula aplicando la Ecuación 5.9.

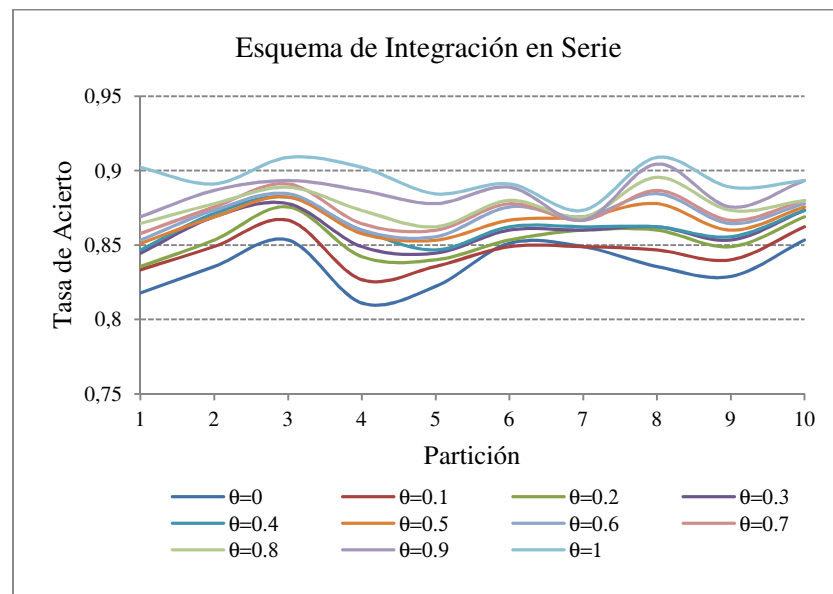
<sup>12</sup> <http://web.engr.oregonstate.edu/~tgd/software/ecoc-codes.tar.gz>

**Tabla 5.34. Código Corrector de Errores para el Reconocimiento de Señales de Tráfico.** La distancia *Hamming* entre filas varía entre 5 y 10. La distancia *Hamming* entre columnas varía entre 2 y 7. La distancia *Hamming* entre columnas complementarias varía entre 2 y 7.

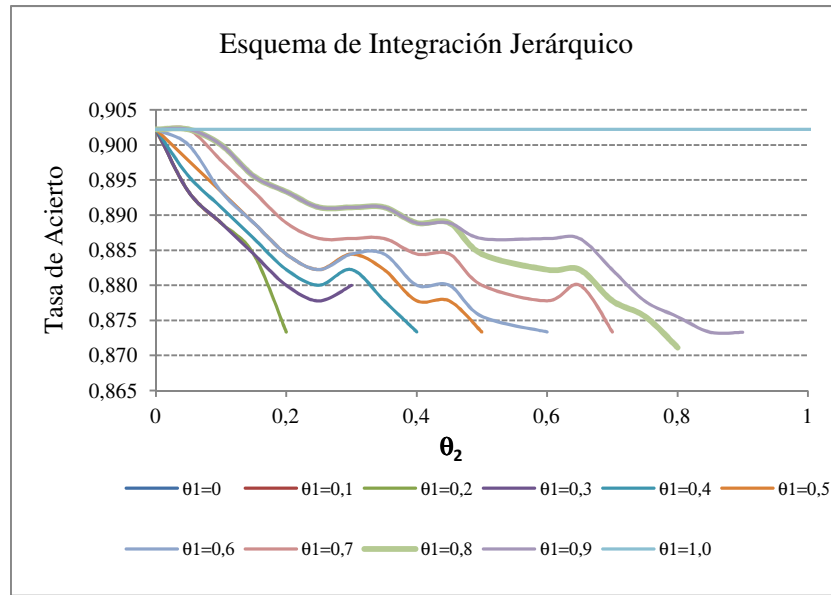
Clase	Código														
	f1	f2	f3	f4	f5	f6	f7	f8	f9	f10	f11	f12	f13	f14	f15
C1	0	1	1	1	0	1	0	1	0	0	1	0	1	1	0
C2	1	0	0	0	0	1	1	1	1	1	0	1	1	1	0
C3	0	1	1	0	1	1	1	1	0	0	0	0	1	0	1
C4	0	1	1	0	1	0	1	0	1	0	1	1	1	0	0
C5	0	1	1	1	0	1	0	0	1	1	0	1	0	0	1
C6	1	0	1	1	0	1	1	0	0	1	1	1	1	0	1
C7	0	0	0	0	0	0	0	1	1	0	1	0	1	0	0
C8	1	1	1	0	0	0	0	1	0	0	0	1	1	0	1
C9	1	0	1	0	1	1	0	0	1	0	1	0	0	1	1

## BCE

Puesto que la implementación de BCE exige conocer el esquema de integración Binario-Complementario usado en sus módulos base, a continuación, y de forma gráfica (Figuras 5.37 y 5.38), se muestran los resultados experimentales obtenidos al evaluar el esquema de integración en serie y el esquema de integración jerárquico sobre este dominio. En este caso y dado que el método experimental contempla 5 iteraciones del proceso de validación cruzada, los valores mostrados reflejan este proceso.



**Figura 5.37. ASISTENTUR: Tasa de Acierto en función de  $\theta$ .**



**Figura 5.38. ASISTENTUR: Tasa de Acierto en función de  $\theta_1$  y  $\theta_2$ .**

Observando los valores recogidos en las Figuras 5.37 y 5.38 se puede comprobar que, de nuevo, el mejor esquema de integración se corresponde con el esquema de integración en paralelo ( $\theta=0$  en el esquema de integración en serie y  $\theta_1=1$  o  $\theta_2=1$  en el esquema de integración jerárquico) por tanto, nuevamente, será este el esquema usado en la implementación de BCE.

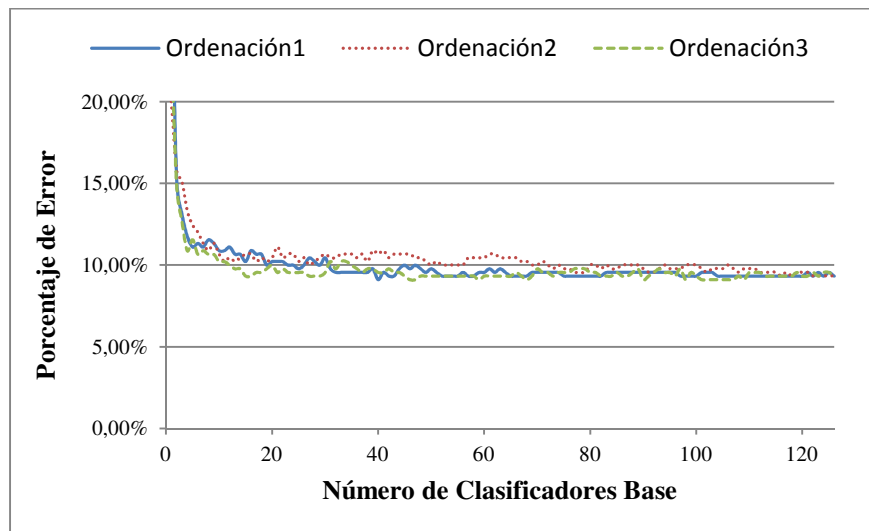
### CCE

Si para la construcción de BCE es necesario determinar el esquema de integración usado en sus módulos base, las restricciones de diseño impuestas sobre CCE implican predeterminar i) el número de módulos base que componen la arquitectura y ii) la matriz de códigos asociada al sistema.

Para reducir al máximo el coste computacional de esta arquitectura y hacerlo comparable al del sistema BCE, se impondrá como condición que, en este caso, el número de módulos coincida con el número de clases. Es decir, fijaremos el número de módulos a 9. Para comprobar los efectos que esta restricción de diseño tiene sobre el comportamiento del conjunto, previo a cualquier otro análisis experimental, se ha analizado la relación que existe entre ambos parámetros. Para ello, y al igual que se hizo con la base de datos *SEMEION* (ver sección 5.2.2) se han establecido todas las posibles combinaciones que engloban a 5 de las 9 clases<sup>13</sup> y, sobre un subconjunto de entrenamiento, se han generado todos los módulos base a los que dan lugar (126). Una vez construidos estos módulos, los mismos se han ordenado de forma totalmente aleatoria y de forma iterativa se ha construido y evaluado (sobre los correspondientes

<sup>13</sup> Las 5 clases seleccionadas determinan los ejemplos de aprendizaje para uno de los clasificadores que integran un módulo base y las 4 clases restantes los ejemplos usados en el aprendizaje del segundo clasificador.

datos de test) el conjunto compuesto por los  $k$  ( $1 \leq k \leq 126$ ) primeros módulos. En la Figura 5.39 se puede ver el resultado de este análisis para tres ordenaciones distintas.



**Figura 5.39. ASISTENTUR: Error en función del número de módulos base de CCE. Los resultados mostrados corresponden a tres ordenaciones distintas de los 126 módulos que se pueden construir.**

Observando los resultados mostrados en la Figura 5.39 se puede comprobar que, para el conjunto de datos seleccionado, el porcentaje de error del sistema cuando éste está formado por 9 módulos es aproximadamente de un 11%. Dado que este valor dista casi un 2% del mínimo absoluto (9,33%) se puede deducir que CCE dispone de un margen de mejora que, en su momento, podrá ser sopesado con respecto al coste computacional que llevaría implícito.

### 5.6.3. Resultados Experimentales

Una vez descritos los modelos de aprendizaje implementados, en este epígrafe se muestran los resultados obtenidos en la evaluación de cada uno de ellos.

Con el propósito de estimar la precisión de los distintos modelos y determinar si las diferencias observadas son significativas, se han realizado 5 réplicas del proceso de *validación cruzada* con 2 particiones y sobre los resultados obtenidos se ha aplicado el *estadístico F*. Además, para evitar que el carácter inestable de las Redes de Neuronas (el modelo resultante puede variar mucho dependiendo del proceso de inicialización), introduzca un sesgo en la comparación estadística, para cada conjunto de entrenamiento se han construido y evaluado 10 versiones de cada uno de los modelos de aprendizaje considerados (cada uno con una inicialización de pesos distinta). De este modo, el error de cada modelo sobre cada conjunto de datos se determinará calculando el número de errores que sobre dicho conjunto de test comete cada una de las distintas versiones que de un mismo modelo se tienen, eliminando los casos extremos y promediando los valores restantes. Así, si los valores mostrados en cada una de las filas de la Tabla 5.35

indican el número de errores que sobre el correspondiente conjunto de test<sup>14</sup> comete una versión específica de un modelo dado, el error del modelo, sobre cada partición, será el indicado en la última fila.

**Tabla 5.35. Número de errores cometido por un modelo de clasificación cuando éste se construye a partir de un determinado subconjunto de datos (Partición  $i$ - $j$ ) y con una determinada configuración inicial (Configuración  $k$ ). La fila inferior (media) muestra el error medio del sistema sobre cada uno de los conjuntos de datos considerados. En el cálculo de la media se excluyen los valores extremos.**

Conf.	Partición 1-1	Partición 1-2	Partición 2-1	Partición 2-2	Partición. 3-1	Partición 3-2	Partición 4-1	Partición 4-2	Partición 5-1	Partición 5-2
1	48	45	41	51	48	51	60	41	46	45
2	43	46	44	50	46	47	55	46	48	49
3	46	48	41	44	46	50	56	45	49	47
4	47	46	42	47	49	50	57	50	48	52
5	46	50	42	48	49	54	60	47	44	49
6	44	47	44	46	45	51	57	47	50	50
7	42	46	41	48	51	48	60	46	51	48
8	45	47	39	46	47	51	57	49	46	48
9	47	44	42	43	48	50	55	46	50	45
10	46	47	42	44	48	50	58	48	48	49
<b>Media</b>	<b>45.50</b>	<b>46.50</b>	<b>41.88</b>	<b>46.63</b>	<b>47.63</b>	<b>50.13</b>	<b>57.50</b>	<b>46.75</b>	<b>48.13</b>	<b>48.13</b>

Para evitar que la distribución de clases usada en la construcción de los módulos base que integran la arquitectura CCE tenga influencia en los resultados experimentales, se admitirá que la topología de cada módulo es un parámetro libre de la arquitectura y, en consecuencia, un parámetro a fijar en cada versión específica del modelo. Por tanto, cada versión de CCE se caracterizará no solo por una inicialización de pesos concreta sino, también, por una distribución de clases específica. De forma análoga, al trabajar con *Bagging*, se considerará que el subconjunto de ejemplos usados en la construcción de cualquier clasificador base es un parámetro libre del modelo, es decir, un factor que puede influir en los resultados experimentales. Por ello, cada versión específica de *Bagging* estará caracterizada por la distribución inicial de pesos y por el subconjunto de ejemplos usado en la construcción de todas y cada una de las redes de neuronas que integran el modelo.

Para facilitar la comparación estadística, el análisis experimental se ha dividido en dos partes. En una primera fase se comparan los resultados de BCE y CCE con los resultados obtenidos al aplicar los modelos de referencia ya indicados (1 red de neuronas -ANN-, arquitectura OAA, *Bagging* y *ECOC*). En la segunda fase se muestran los resultados obtenidos cuando los clasificadores de referencia se construyen a partir de ejemplos sobre los que se ha realizado un proceso de selección de atributos.

Las Tablas 5.36 y 5.37 recogen los valores experimentales obtenidos en la primera fase de la experimentación.

<sup>14</sup> Conjunto de datos resultante de repetir 5 veces el proceso de validación cruzada con dos particiones.

**Tabla 5.36. Resultados experimentales obtenidos al evaluar los distintos modelos construidos. En las nueve primeras filas se muestra el porcentaje de ejemplos correctamente clasificados para cada una de las clases. En la última fila se muestra el porcentaje de ejemplos correctamente clasificados respecto al total de ejemplos. En negrita se indican los mejores valores.**

	1 ANN	OAA	Bagging	ECOC	BCE	CCE
C1	98,30%	98,40%	<b>98,90%</b>	98,78%	98,20%	98,25%
C2	98,53%	98,98%	<b>99,60%</b>	98,73%	99,15%	99,50%
C3	89,55%	90,00%	<b>94,90%</b>	93,08%	89,15%	91,28%
C4	91,73%	92,40%	93,05%	<b>94,33%</b>	94,10%	92,10%
C5	95,30%	94,95%	95,80%	<b>95,88%</b>	95,40%	94,40%
C6	73,45%	74,30%	76,08%	74,30%	74,35%	<b>77,15%</b>
C7	81,23%	82,10%	81,98%	<b>82,45%</b>	82,40%	81,80%
C8	88,03%	88,10%	89,18%	91,15%	90,98%	<b>91,53%</b>
C9	84,55%	85,08%	<b>87,58%</b>	86,70%	80,53%	82,50%
Global	88,96%	89,37%	<b>90,78%</b>	90,60%	89,36%	89,83%

**Tabla 5.37. Número de errores cometido por los distintos modelos al ejecutar el proceso de validación cruzada con 5 réplicas y 2 particiones. El número de ejemplos evaluados en cada ejecución es 450. La última fila muestra la media de los errores cometidos por cada modelo. En negrita se indican los mejores valores.**

Réplica-Partición	1 ANN	OAA	Bagging	ECOC	BCE	CCE
1-1	51,50	48,88	<b>41,13</b>	42,38	45,50	48,63
1-2	45,00	43,00	<b>34,75</b>	39,00	46,50	39,00
2-1	44,75	42,50	<b>38,00</b>	39,25	41,88	40,13
2-2	55,50	51,88	<b>44,00</b>	<b>44,00</b>	46,63	47,63
3-1	51,38	51,63	48,25	50,38	<b>47,63</b>	50,50
3-2	48,75	47,75	<b>38,38</b>	40,63	50,13	48,38
4-1	50,38	50,25	<b>40,50</b>	48,50	57,50	54,88
4-2	50,13	46,63	42,13	<b>39,38</b>	46,75	43,50
5-1	54,50	52,13	47,13	43,25	48,13	<b>42,63</b>
5-2	44,88	43,88	40,50	<b>36,38</b>	48,13	42,25
Media	49,67	47,85	<b>41,48</b>	42,31	47,88	45,75

La Tabla 5.38. recoge los resultados obtenidos al aplicar el estadístico *F-test* sobre los resultados mostrados en la Tabla 5.37. Analizando estos valores, se puede concluir con un nivel de confianza del 95% que los modelos propuestos en este trabajo –BCE y CCE- son estadísticamente equivalentes entre sí ( $F=1,15$ ) y estadísticamente equivalentes al resto de modelos evaluados: En todos los casos el valor del *estadístico F* es menor que 4,735 (valor del *estadístico F* con 5 y 10 grados de libertad y un nivel de significancia de 0,05).



**Tabla 5.38.** Comparación estadísticas de los modelos implementados. El símbolo (.) indica que el modelo M1 (horizontal) es estadísticamente equivalente al modelo M2 (vertical). El símbolo (+)/(-) indica que el modelo M1 (horizontal) es estadísticamente mejor/peor que el modelo M2 (vertical).

	1 ANN	OAA	Bagging	ECOC	BCE	CCE
1 ANN	X	2,87 (.)	8,49 (-)	3,65 (.)	0,79 (.)	1,48 (.)
OAA		X	5,94 (-)	3,97 (.)	0,70 (.)	1,16 (.)
Bagging			X	1,06 (.)	1,69 (.)	1,53 (.)
ECOC				X	2,17 (.)	1,35 (.)
BCE					X	1,15 (.)
CCE						X

Sin embargo, y como se refleja en la Tabla 5.39, la drástica disminución en el número de características que describen los ejemplos hace que tanto BCE como CCE sean mucho mejores en lo que a tamaño, complejidad computacional y tiempo de cálculo se refiere.

**Tabla 5.39.** Tiempos de clasificación y aprendizaje de cada clasificador base medidos con el comando *time* de *Linux* en un Ordenador Intel(R) Core(TM) i7-2600 CPU @ 3.40GHz y detalles de la arquitectura de los métodos implementados. Para BCE y CCE se indican los valores para cada uno de los clasificadores que integran un módulo. Como número de entradas se refleja el valor medio.

	Tiempo de clasificación (450 ejem.)	Tiempo de aprendizaje (2000 ciclos)	N. de módulos	Neuronas entrada	Pesos (Capa 1)	Neuronas ocultas	Pesos (Capa 2)	Neur. de salida
1 ANN	0,190s	2m 44,870s	1	1024	51200	50	450	9
OAA	0,480s	1m 03,580s	9	1024	51200	50	50	1
Bagging	0,850s	2m 43,420s	20	1024	51200	50	450	9
ECOC	0,680s	2m 45,420s	15	1024	51200	50	50	1
BCE	0,290s	13,480s	9	74	2220	30	30	1
		10,780s	9	131	3930	30	240	8
CCE	0,220s	9,070s	9	109	3270	30	150	5
		5,430s	9	121	3630	30	120	4

Para comprobar si los modelos propuestos en este trabajo son una buena alternativa para la construcción de sistemas que han de trabajar en tiempo real, en la segunda fase de la experimentación, se ha optado por analizar la repercusión que sobre el resto de modelos estudiados tiene el proceso de selección de atributos. Dado que cada uno de los clasificadores base que integran *OAA*, *Bagging* y *ECOC* se construye a partir de un determinado subconjunto de ejemplos el primer aspecto a analizar es si el proceso de selección de características debe realizarse antes o después de generar estos subconjuntos (ver Figura 5.29).

En la Tabla 5.40 se presentan los resultados experimentales obtenidos cuando cada uno de los modelos indicados se construye:

- a) usando un único subconjunto de características,
- b) usando un conjunto de características específico para cada clasificador base.

**Tabla 5.40. Errores obtenidos al evaluar el método de selección de características sobre OAA, Bagging y ECOC.**

	OAA		Bagging		ECOC	
	Selección única	Selección múltiple	Selección única	Selección múltiple	Selección única	Selección múltiple
Cjto. 1-1	<b>58,38</b>	77,75	57,38	<b>49,38</b>	55,00	<b>50,13</b>
Cjto. 1-2	<b>56,88</b>	69,88	54,00	<b>45,13</b>	55,75	<b>50,38</b>
Cjto. 2-1	<b>51,38</b>	63,25	47,13	<b>43,00</b>	<b>47,75</b>	48,50
Cjto. 2-2	<b>58,75</b>	83,38	55,38	<b>47,88</b>	54,63	<b>51,00</b>
Cjto. 3-1	<b>61,50</b>	81,75	59,00	<b>52,88</b>	57,25	<b>54,13</b>
Cjto. 3-2	<b>58,38</b>	69,50	53,00	<b>50,25</b>	59,63	<b>49,13</b>
Cjto. 4-1	68,38	<b>66,75</b>	63,63	<b>59,00</b>	68,38	<b>59,63</b>
Cjto. 4-2	<b>57,00</b>	77,63	53,63	<b>44,38</b>	<b>50,00</b>	56,00
Cjto. 5-1	<b>62,25</b>	77,00	54,25	<b>49,63</b>	53,63	<b>49,63</b>
Cjto. 5-2	<b>58,88</b>	68,38	54,25	<b>50,75</b>	<b>47,38</b>	53,00
Media	<b>59,18</b>	73,53	55,16	<b>49,23</b>	54,94	<b>52,15</b>
F_Test	3,166		8,694		0,903	

Observando los valores recogidos en la Tabla 5.40 se comprueba que cuando los clasificadores base se construyen usando un único subconjunto de características, la precisión de OAA mejora mientras que la de Bagging y ECOC empeora. No obstante, al aplicar el *estadístico F* (columna inferior de la Tabla 5.39) se comprueba que las diferencias observadas solo son estadísticamente significativas en el caso de *Bagging* mientras que *OAA* y *ECOC* parecen ser invariables al punto en el que se realiza el proceso de selección de atributos. A pesar de las equivalencias estadísticas encontradas, a la hora de comparar *OAA* y *ECOC* con los modelos propuestos en este trabajo -BCE y CCE-, únicamente se tendrán en consideración aquellas versiones que ofrecen una menor tasa de error. Es decir, en lo que sigue, usaremos la versión de *OAA* construida a partir de un único conjunto de características y la versión de *Bagging* y *ECOC* en la que cada clasificador base se construye a partir de un subconjunto de características específico.

En las Tablas 5.41 y 5.42 se muestran los resultados obtenidos al evaluar los distintos conjuntos de clasificadores (1 Red de neuronas, *OAA*, *Bagging*, *ECOC*, BCE y CCE) cuando éstos se construyen a partir de ejemplos sobre los que se ha realizado un proceso de selección de atributos. Observando los valores recogidos en dichas tablas se puede comprobar que, en el dominio analizado, los modelos de clasificación que ofrecen mejores resultados son CCE y BCE. Además, al analizar los resultados recogidos en la Tabla 5.41 destaca la considerable mejora que, respecto a la clase más difícil de aprender, clase C6, introduce CCE respecto al resto de modelos (la tasa de acierto de CCE al considerar esta clase aumenta un 9% respecto al valor obtenido por *Bagging* o *AOO* y un 5% respecto al valor obtenido con *ECOC*).

Tabla 5.41. Resultados experimentales obtenidos por los distintos clasificadores cuando éstos se combinan con un proceso de selección de atributos. En las nueve primeras filas se muestra el porcentaje de ejemplos correctamente clasificados para cada una de las clases. En la última fila se muestra el porcentaje de ejemplos correctamente clasificados respecto al total de ejemplos. En negrita se indican los mejores valores.

	1 ANN	OAA	Bagging	ECOC	BCE	CCE
C1	98,25%	98,23%	98,53%	<b>98,73 %</b>	98,20%	98,25%
C2	98,20%	98,53%	99,18%	<b>99,63 %</b>	99,15%	99,50%
C3	84,08%	82,93%	88,28%	89,13%	89,15%	<b>91,28 %</b>
C4	92,98%	93,88%	91,85%	93,68%	<b>94,10 %</b>	92,10%
C5	94,23%	94,18%	94,43%	<b>95,40 %</b>	<b>95,40 %</b>	94,40%
C6	69,50%	68,43%	68,15%	72,33%	74,35%	<b>77,15 %</b>
C7	80,53%	80,75%	79,63%	80,83%	<b>82,40 %</b>	81,80%
C8	88,45%	88,35%	90,03%	90,60%	90,98%	<b>91,53 %</b>
C9	76,80%	76,40%	79,98%	81,25%	80,53%	<b>82,50 %</b>
Global	87,00%	86,85%	87,78%	89,06%	89,36%	<b>89,83 %</b>

Tabla 5.42. Número de errores cometido por los distintos modelos cuando éstos se combinan con un proceso de selección de atributos. Los errores mostrados se corresponden con los obtenidos al ejecutar el proceso de validación cruzada con 5 réplicas y 2 particiones. El número de ejemplos evaluados en cada ejecución es 450. La última fila muestra la media de los errores cometidos por cada modelo. En negrita se indican los mejores valores.

Réplica-Partición	1 ANN	OAA	Bagging	ECOC	BCE	CCE
1-1	59,13	58,38	49,38	50,13	<b>45,50</b>	48,63
1-2	60,13	56,88	45,13	50,38	46,50	<b>39,00</b>
2-1	52,88	51,38	43,00	48,50	41,88	<b>40,13</b>
2-2	58,88	58,75	47,88	51,00	<b>46,63</b>	47,63
3-1	60,25	61,50	52,88	54,13	<b>47,63</b>	50,50
3-2	58,88	58,38	50,25	49,13	50,13	<b>48,38</b>
4-1	65,13	68,38	59,00	59,63	57,50	<b>54,88</b>
4-2	55,25	57,00	44,38	56,00	46,75	<b>43,50</b>
5-1	57,88	62,25	49,63	49,63	48,13	<b>42,63</b>
5-2	56,63	58,88	50,75	53,00	48,13	<b>42,25</b>
Media	58,50	59,18	49,23	52,15	47,88	<b>45,75</b>

Sin embargo, al aplicar el *estadístico F* (Tabla 5.43) sobre los valores mostrados en la Tabla 5.42 se observa que, a pesar de las diferencias encontradas, *Bagging*, *ECOC*, *BCE* y *CCE* son estadísticamente equivalentes entre sí.

**Tabla 5.43. Comparación estadísticas de los modelos implementados. El símbolo (+)/(-) indica que el modelo M1 (horizontal) es significativamente mejor/peor que el modelo M2 (vertical). El símbolo (.) indica que entre los modelos M1 y M2 no hay diferencias significativas.**

	1 ANN	OAA	Bagging	ECOC	BCE	CCE
1ANN	X	2,95 (.)	15,83 (-)	5,81 (-)	62,37 (-)	14,59 (-)
OAA		X	22,85 (.)	4,34 (.)	23,20 (-)	25,14 (-)
Bagging			X	1,35 (.)	0,92 (-)	3,99 (.)
ECOC				X	2,09 (.)	2,78 (.)
BCE					X	1,15 (.)
CCE						X

Para comprobar si existe una relación entre la diversidad de los clasificadores base y la precisión de los distintos modelos construidos, en la Tabla 5.44 se recogen los valores de precisión y diversidad obtenidos al evaluar Bagging, BCE y CCE<sup>15</sup>.

---

<sup>15</sup> El cálculo de diversidad solo se realiza sobre aquellos modelos en los que los clasificadores base son redundantes, es decir, sobre aquellos modelos en los que cada clasificadores base aporta, por sí mismo, una solución no ambigua al problema de clasificación. Por tanto, junto a la red de neuronas simple (que no es estrictamente un conjunto de clasificadores), en el cálculo de la diversidad se excluyen la arquitectura OAA y ECOC pues, en ambos casos, la salida dada por uno solo de sus clasificadores base no constituye una solución al problema de clasificación..

**Tabla 5.44. Valores de diversidad y precisión para Bagging, BCE y CCE. En negrita se indican los mejores valores.**

Réplica/ Partición	Modelo	$f(\uparrow)$	$des(\uparrow)$	$Q(\downarrow)$	$\rho(\downarrow)$	$\kappa(\downarrow)$	$amb(\uparrow)$	Precisión
1-1	<i>Bagging_1024</i>	0,1751	0,1332	0,8812	0,5019	0,8243	0,0185	<b>90,86%</b>
	<i>Bagging_BFCFS</i>	0,1701	0,1306	0,8960	0,5331	0,8292	0,0180	89,03%
	BCE	0,1542	0,1217	0,9025	0,5279	0,8453	0,0152	89,89%
	CCE	<b>0,3829</b>	<b>0,2862</b>	<b>0,6311</b>	<b>0,3305</b>	<b>0,6112</b>	<b>0,0378</b>	89,19%
1-2	<i>Bagging_1024</i>	0,1508	0,1168	0,8965	0,5084	0,8488	0,0159	<b>92,28%</b>
	<i>Bagging_BFCFS</i>	0,1614	0,1201	0,9063	0,5431	0,8381	0,0170	89,97%
	BCE	0,1477	0,1094	0,9211	0,5681	0,8519	0,0146	89,67%
	CCE	<b>0,3476</b>	<b>0,2644</b>	<b>0,6568</b>	<b>0,3396</b>	<b>0,6471</b>	<b>0,0343</b>	91,33%
2-1	<i>Bagging_1024</i>	0,1644	0,1183	0,8973	0,5133	0,8350	0,0174	<b>91,56%</b>
	<i>Bagging_BFCFS</i>	0,1604	0,1148	0,9090	0,5399	0,8391	0,0169	90,44%
	BCE	0,1437	0,1054	0,9185	0,5517	0,8560	0,0142	90,69%
	CCE	<b>0,3606</b>	<b>0,2764</b>	<b>0,6233</b>	<b>0,3135</b>	<b>0,6342</b>	<b>0,0356</b>	91,08%
2-2	<i>Bagging_1024</i>	0,1724	0,1312	0,8833	0,5040	0,8270	0,0182	<b>90,22%</b>
	<i>Bagging_BFCFS</i>	0,1742	0,1303	0,8948	0,5319	0,8251	0,0184	89,36%
	BCE	0,1665	0,1284	0,8905	0,5148	0,8330	0,0164	89,64%
	CCE	<b>0,3688</b>	<b>0,2754</b>	<b>0,6451</b>	<b>0,3355</b>	<b>0,6250</b>	<b>0,0364</b>	89,42%
3-1	<i>Bagging_1024</i>	0,1660	0,1181	0,9160	0,5656	0,8334	0,0175	89,28%
	<i>Bagging_BFCFS</i>	0,1657	0,1160	0,9214	0,5803	0,8337	0,0175	88,25%
	BCE	0,1496	0,1094	0,9245	0,5742	0,8500	0,0148	<b>89,42%</b>
	CCE	<b>0,3557</b>	<b>0,2629</b>	<b>0,6790</b>	<b>0,3606</b>	<b>0,6387</b>	<b>0,0351</b>	88,78%
3-2	<i>Bagging_1024</i>	0,1580	0,1126	0,9089	0,5336	0,8415	0,0167	<b>91,47%</b>
	<i>Bagging_BFCFS</i>	0,1702	0,1237	0,9055	0,5470	0,8291	0,0180	88,83%
	BCE	0,1549	0,1175	0,9139	0,5496	0,8447	0,0153	88,86%
	CCE	<b>0,3436</b>	<b>0,2503</b>	<b>0,7101</b>	<b>0,3849</b>	<b>0,6519</b>	<b>0,0339</b>	89,25%
4-1	<i>Bagging_1024</i>	0,1701	0,1263	0,8898	0,5088	0,8293	0,0180	<b>91,00%</b>
	<i>Bagging_BFCFS</i>	0,1704	0,1243	0,9108	0,5637	0,8289	0,0180	86,89%
	BCE	0,1515	0,1101	0,9300	0,5923	0,8481	0,0150	87,22%
	CCE	<b>0,3435</b>	<b>0,2568</b>	<b>0,7010</b>	<b>0,3751</b>	<b>0,6515</b>	<b>0,0339</b>	87,80%
4-2	<i>Bagging_1024</i>	0,1605	0,1221	0,8944	0,5135	0,8390	0,0169	<b>90,64%</b>
	<i>Bagging_BFCFS</i>	0,1635	0,1197	0,9092	0,5485	0,8360	0,0173	90,14%
	BCE	0,1517	0,1154	0,9082	0,5323	0,8479	0,0150	89,61%
	CCE	<b>0,3663</b>	<b>0,2828</b>	<b>0,6133</b>	<b>0,3103</b>	<b>0,6275</b>	<b>0,0362</b>	90,33%
5-1	<i>Bagging_1024</i>	0,1701	0,1277	0,8936	0,5222	0,8292	0,0180	<b>89,53%</b>
	<i>Bagging_BFCFS</i>	0,1765	0,1332	0,8897	0,5223	0,8228	0,0186	88,97%
	BCE	0,1660	0,1271	0,8974	0,5307	0,8335	0,0164	89,30%
	CCE	<b>0,3578</b>	<b>0,2705</b>	<b>0,6430</b>	<b>0,3320</b>	<b>0,6368</b>	<b>0,0353</b>	90,53%
5-2	<i>Bagging_1024</i>	0,1618	0,1229	0,8889	0,5017	0,8376	0,0171	<b>91,00%</b>
	<i>Bagging_BFCFS</i>	0,1586	0,1156	0,9153	0,5594	0,8409	0,0167	88,72%
	BCE	0,1543	0,1146	0,9126	0,5497	0,8453	0,0152	89,30%
	CCE	<b>0,3549</b>	<b>0,2660</b>	<b>0,6560</b>	<b>0,3400</b>	<b>0,6400</b>	<b>0,0351</b>	90,61%

Analizando los valores recogidos en la Tabla 5.44 se comprueba que los mejores valores en lo que a diversidad se refiere se consiguen con CCE mientras que los mejores valores de precisión se logran con *Bagging* cuando éste se construye usando las 1024 características que describen los ejemplos. Por tanto, basándonos en estos valores, resulta difícil establecer una relación entre los parámetros considerados.

Dado que, la diversidad es una medida que pretende cuantificar las diferencias existentes entre los distintos clasificadores base y la influencia que dicha diferencia tiene en la precisión del conjunto se ha optado por realizar un estudio algo más exhaustivo. En este segundo estudio se analiza la relación existente entre las distintas medidas de diversidad y la mejora que, con respecto a la precisión media de sus miembros, se consigue al usar un determinado modelo. Es decir, se analiza la relación *diversidad vs (P<sub>CONJUNTO</sub> - P<sub>MEDIA</sub>)*. Aunque esta relación podría establecerse en términos del *Coeficiente de Correlación de Pearson* o del *Coeficiente de Correlación de Spearman* [Tsymbal et al., 2003], en este trabajo hemos optado por establecerla en términos de la *proporción de variabilidad no explicada* [Spiegel, 1976]. Así, si  $r$  es el coeficiente de *Correlación de Pearson* entre  $x$  e  $y$ , e  $y_{est}$  es el valor estimado de  $y$  para un valor dado de  $x$  obtenido de la curva de regresión de  $y$  sobre  $x$ , entonces, por definición:

$$1 - r^2 = \frac{\sum(y - y_{est})^2}{\sum(y - \bar{y})^2} = \frac{\text{variación no explicada}}{\text{variación total}} \quad [\text{Ec. 5.23}]$$

representa la proporción de casos en los que el valor de  $y$  no puede explicarse en términos del valor de  $x$ . En el caso que nos ocupa,  $r^2$  representará la proporción de casos en los que la mejora del conjunto puede explicarse en términos de diversidad y  $1 - r^2$  (coeficiente de no determinación) representará el número de casos en los que la mejora del conjunto queda sin explicar. Por tanto, cuanto menor sea el valor de  $(1 - r^2)$  mayor será el porcentaje de casos en los que la mejora del conjunto se puede considerar consecuencia de la diversidad inducida en los clasificadores base. En la Tabla 5.45 se recoge el valor del coeficiente de no determinación  $(1 - r^2)$  obtenido para cada una de las medidas de diversidad usadas. A título informativo, también se muestra el valor medio de mejora del conjunto  $(P_{CONJUNTO} - P_{MEDIA})$  en cada caso.

Observando los valores recogidos en la se puede comprobar que, salvo excepciones, el menor porcentaje de casos no explicados ocurre en el modelo CCE.

Tabla 5.45. Coeficiente de no determinación ( $1-r^2$ ) entre las distintas medidas de diversidad calculadas ( $f$ ,  $des$ ,  $Q$ ,  $\rho$ ,  $\kappa$ ,  $amb$ ) y la Mejora del conjunto ( $P_{CJTO}-P_{MEDIA}$ ) para cada uno de los modelos analizados.

Réplica/ Partición	Modelo	$f$	$des$	$Q$	$\rho$	$\kappa$	$amb$	Mejora
1-1	Bagging_1024	95,33%	100,00%	95,10%	89,04%	95,34%	95,33%	6,70%
	Bagging_BFCFS	56,80%	61,64%	74,29%	83,19%	56,83%	56,78%	5,77%
	BCE	54,37%	73,67%	93,40%	98,34%	54,49%	54,35%	4,91%
	CCE	<b>22,70%</b>	<b>26,51%</b>	<b>39,75%</b>	<b>53,10%</b>	<b>23,82%</b>	<b>22,70%</b>	<b>19,58%</b>
1-2	Bagging_1024	55,39%	62,21%	58,88%	59,45%	55,37%	55,34%	5,98%
	Bagging_BFCFS	62,47%	42,02%	<b>30,60%</b>	<b>35,67%</b>	62,49%	62,45%	5,46%
	BCE	32,78%	26,87%	47,45%	86,09%	32,77%	32,79%	2,59%
	CCE	<b>15,16%</b>	<b>20,05%</b>	53,89%	68,37%	<b>14,93%</b>	<b>15,15%</b>	<b>18,38%</b>
2-1	Bagging_1024	94,05%	86,73%	91,61%	94,84%	94,09%	94,05%	5,60%
	Bagging_BFCFS	52,49%	61,70%	<b>53,25%</b>	75,67%	52,41%	52,51%	4,98%
	BCE	52,71%	67,07%	79,62%	74,92%	52,72%	52,67%	3,34%
	CCE	<b>24,98%</b>	<b>30,03%</b>	65,99%	72,40%	<b>25,36%</b>	<b>24,98%</b>	<b>18,41%</b>
2-2	Bagging_1024	93,36%	89,92%	84,64%	87,49%	93,36%	93,35%	5,87%
	Bagging_BFCFS	41,78%	38,82%	44,69%	57,07%	41,75%	41,77%	6,00%
	BCE	49,30%	57,20%	90,02%	99,88%	49,26%	49,30%	3,70%
	CCE	<b>8,77%</b>	<b>20,38%</b>	<b>39,72%</b>	<b>54,11%</b>	<b>8,41%</b>	<b>8,77%</b>	<b>18,18%</b>
3-1	Bagging_1024	44,59%	51,45%	65,74%	79,86%	44,56%	44,57%	5,42%
	Bagging_BFCFS	68,20%	56,96%	64,40%	76,35%	68,28%	68,17%	4,80%
	BCE	52,95%	70,03%	95,31%	99,66%	52,92%	52,97%	3,59%
	CCE	<b>9,29%</b>	<b>21,28%</b>	<b>45,05%</b>	<b>48,61%</b>	<b>9,17%</b>	<b>9,29%</b>	<b>17,12%</b>
3-2	Bagging_1024	98,06%	99,06%	95,46%	95,54%	98,08%	98,07%	5,47%
	Bagging_BFCFS	54,57%	59,27%	71,86%	80,25%	54,68%	54,56%	5,00%
	BCE	86,63%	79,67%	85,39%	93,53%	86,69%	86,63%	3,22%
	CCE	<b>24,89%</b>	<b>32,17%</b>	<b>57,66%</b>	<b>64,75%</b>	<b>24,77%</b>	<b>24,89%</b>	<b>17,17%</b>
4-1	Bagging_1024	67,57%	65,14%	63,01%	64,16%	67,58%	67,58%	6,12%
	Bagging_BFCFS	79,08%	88,47%	85,55%	91,51%	79,05%	79,06%	4,04%
	BCE	58,18%	52,40%	72,53%	80,60%	58,22%	58,22%	2,84%
	CCE	<b>6,24%</b>	<b>9,83%</b>	<b>28,58%</b>	<b>40,13%</b>	<b>6,10%</b>	<b>6,24%</b>	<b>16,06%</b>
4-2	Bagging_1024	78,65%	70,32%	<b>64,59%</b>	<b>65,00%</b>	78,63%	78,64%	5,35%
	Bagging_BFCFS	<b>25,61%</b>	51,98%	82,58%	96,21%	<b>25,80%</b>	<b>25,60%</b>	5,77%
	BCE	52,74%	<b>50,66%</b>	75,69%	92,51%	52,70%	52,75%	2,84%
	CCE	90,27%	81,24%	91,20%	97,31%	89,72%	90,26%	<b>18,64%</b>
5-1	Bagging_1024	79,74%	94,45%	99,15%	92,49%	79,84%	79,74%	5,38%
	Bagging_BFCFS	69,86%	77,75%	87,32%	90,79%	69,78%	69,86%	5,65%
	BCE	80,41%	40,53%	77,51%	74,07%	80,49%	80,42%	4,36%
	CCE	<b>19,51%</b>	<b>26,77%</b>	<b>60,88%</b>	<b>73,12%</b>	<b>18,59%</b>	<b>19,52%</b>	<b>18,41%</b>
5-2	Bagging_1024	64,06%	76,18%	86,00%	97,63%	64,22%	64,03%	5,43%
	Bagging_BFCFS	100,00%	99,68%	98,25%	98,33%	100,00%	100,00%	4,23%
	BCE	44,31%	63,69%	84,95%	<b>91,99%</b>	44,20%	44,32%	3,35%
	CCE	<b>10,06%</b>	<b>26,23%</b>	<b>79,77%</b>	100,00%	<b>10,46%</b>	<b>10,06%</b>	<b>18,14%</b>

#### 5.6.4. Efecto del Ruido

Uno de los aspectos que deberían estar presentes a la hora de validar cualquier modelo de clasificación es el análisis de la influencia que el ruido tiene en el comportamiento del sistema. Algunos autores se han centrado en estudiar la robustez de los modelos cuando tanto el conjunto de entrenamiento como el de test contienen ejemplos incorrectamente etiquetados [Dietterich 200b, García-Pedrajas, et al., 2007]. sin embargo, a nuestro entender, consideramos que resulta más interesante estudiar la bondad del sistema cuando el ruido está presente únicamente en los ejemplos de aprendizaje. Así pues, para analizar el efecto del ruido seguiremos un método experimental similar al propuesto en [Dietterich, 2000b] el cual implica seleccionar de forma aleatoria un determinado porcentaje de ejemplos,  $r$ , y cambiar su clase para conseguir que ésta sea incorrecta. Sin embargo, a diferencia de este autor y aunque el porcentaje de error,  $r$ , será aplicado sobre todos los ejemplos disponibles, en la fase de test el subconjunto de ejemplos usado estará libre de ruido. En la Figura 5.40 se esquematiza la inyección de ruido cuando el procedimiento usado en la construcción del clasificador sigue un proceso de *validación cruzada* con 2 particiones.

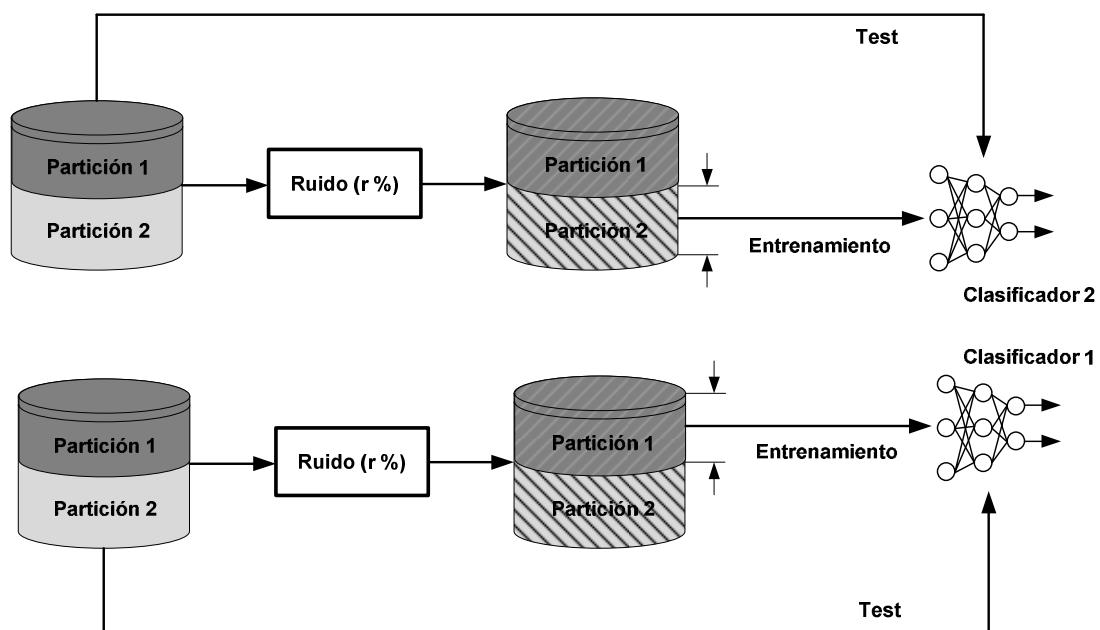


Figura 5.40. Evaluación del ruido. Esquema seguido en la construcción de los clasificadores.

Para analizar la influencia que tiene el ruido en el comportamiento de los modelos propuestos en este trabajo, se seguirá un proceso experimental similar al descrito en el epígrafe 5.6.1. Es decir, se realizarán 5 iteraciones del proceso de validación de 2 particiones. Además, los resultados obtenidos se compararán, mediante el *estadístico F*, con los obtenidos al usar *Bagging* y *ECOC* cuando éstos se construyen a partir de ejemplos sobre los que se ha realizado un proceso de selección de atributos.

Por analogía con los trabajos que estudian la influencia del ruido en el comportamiento de los sistemas, el nivel de ruido inducido será del 10% y del 20% y se garantizará que



el ruido está equitativamente distribuido entre las distintas clases. Además, para reducir el tiempo de cómputo, en este caso, se admitirá que los parámetros libres de los distintos modelos implementados no influyen en los resultados obtenidos. Por tanto, para cada conjunto de entrenamiento y para cada nivel de ruido sólo se construirá una versión de cada uno de los sistemas indicados. En las Tablas 5.46 y 5.47 se recogen los resultados obtenidos tras la fase experimental.

**Tabla 5.46. Precisión por clases y precisión global dada por los distintos clasificadores cuando se evalúa su robustez frente al ruido. En la parte de la izquierda se muestran los resultados obtenidos cuando el nivel de ruido inducido es del 10%. En la parte de la derecha se muestran los resultados obtenidos cuando el nivel de ruido inducido es del 20%. En negrita se indican los mejores valores.**

	ECOC	Bagging	BCE	CCE	ECOC	Bagging	BCE	CCE
C1	98,60%	98,60%	<b>98,80%</b>	98,60%	<b>98,80%</b>	98,40%	98,00%	98,00%
C2	98,60%	98,40%	99,20%	<b>100,00%</b>	97,80%	98,20%	98,20%	<b>98,40%</b>
C3	84,80%	86,20%	86,60%	<b>89,60%</b>	81,00%	<b>85,20%</b>	84,80%	<b>85,20%</b>
C4	93,20%	94,20%	<b>94,40%</b>	93,20%	<b>92,80%</b>	90,60%	90,80%	89,60%
C5	93,60%	<b>95,00%</b>	<b>95,00%</b>	93,80%	87,60%	<b>92,60%</b>	91,20%	91,00%
C6	69,00%	72,00%	73,60%	<b>75,40%</b>	70,00%	69,80%	71,20%	<b>72,20%</b>
C7	77,80%	79,40%	<b>80,40%</b>	79,80%	76,80%	80,20%	<b>81,40%</b>	80,40%
C8	86,80%	89,80%	<b>90,40%</b>	90,00%	86,80%	88,20%	85,60%	<b>88,60%</b>
C9	76,80%	<b>79,60%</b>	78,20%	<b>79,60%</b>	74,40%	77,80%	76,80%	<b>78,00%</b>
Global	86,58%	88,13%	88,51%	<b>88,89%</b>	85,11%	86,78%	86,44%	<b>86,82%</b>

10%
20%

**Tabla 5.47. Número de errores cometido por los distintos modelos cuando se evalúa su robustez frente al ruido. En la parte de la izquierda se muestran los resultados obtenidos cuando el nivel de ruido inducido es del 10%. En la parte de la derecha se muestran los resultados obtenidos cuando el nivel de ruido inducido es del 20%. En negrita se indican los mejores valores.**

Réplica/ Partición	ECOC	Bagging	BCE	CCE	ECOC	Bagging	BCE	CCE
1-1	62	55	<b>49</b>	55	72	<b>67</b>	74	70
1-2	<b>44</b>	50	50	48	62	54	53	<b>44</b>
2-1	66	42	47	<b>40</b>	67	60	<b>59</b>	62
2-2	71	52	<b>48</b>	53	73	<b>52</b>	61	54
3-1	67	65	<b>58</b>	61	73	58	62	<b>53</b>
3-2	53	52	51	<b>46</b>	58	52	60	<b>49</b>
4-1	66	<b>60</b>	62	<b>60</b>	71	<b>65</b>	66	71
4-2	46	<b>42</b>	45	45	62	<b>54</b>	57	58
5-1	70	57	52	<b>48</b>	68	66	<b>55</b>	62
5-2	59	59	55	<b>44</b>	64	67	<b>63</b>	70
Media	60,4	53,4	51,7	<b>50,0</b>	67,0	59,5	61,0	<b>59,3</b>

10%
20%

Observando los valores mostrados en las Tablas 5.46 y 5.47 se puede comprobar que el sistema que parece tener un mejor comportamiento frente al ruido es CCE. Sin embargo, al evaluar la equivalencia estadística (Tabla 5.48) se comprueba que CCE sólo supera a *ECOC* y *Bagging* cuando el nivel de ruido inducido es del 10%. En el resto de casos, y siempre según los resultados dados por el estadístico *F*, las diferencias observadas parecen no ser estadísticamente significativas.

**Tabla 5.48. Comparación estadística de los modelos implementados cuando se evalúa su robustez frente al ruido. En la parte de la izquierda se muestran los resultados obtenidos cuando el nivel de ruido inducido es del 10%. En la parte de la derecha se muestran los resultados obtenidos cuando el nivel de ruido inducido es del 20%. El símbolo (+)/(-) indica que el modelo M1 (horizontal) es significativamente mejor/peor que el modelo M2 (vertical). El símbolo (.) indica que entre los modelos M1 y M2 no hay diferencias significativas.**

	ECOC	Bagging	BCE	CCE	ECOC	Bagging	BCE	CCE
ECOC	X	3,39 (.)	2,44 (-)	<b>7,29 (-)</b>	X	3,02 (.)	1,42 (.)	1,75 (.)
Bagging		X	1,17 (.)	<b>6,04 (-)</b>		X	1,54 (.)	1,01 (.)
BCE			X	0,89 (.)			X	3,32 (.)
CCE				X				X

### 5.6.5. Conclusiones

Los resultados obtenidos al analizar el comportamiento de BCE y CCE reflejan que ambos modelos son una buena alternativa para el reconocimiento de señales de tráfico. El estudio realizado indica que, en lo que a precisión se refiere, ambos modelos son estadísticamente equivalentes a *Bagging* y a *ECOC* pero mejores en cuanto a tamaño, complejidad computacional y tiempo de cálculo.

Al comparar los modelos propuestos en este trabajo con *Bagging* y *ECOC* cuando éstos se construyen partiendo de ejemplos sobre los que se ha realizado un proceso de características, el estudio experimental realizado indica que el comportamiento de los cuatro modelos vuelve a ser estadísticamente equivalente. No obstante, al realizar un análisis por clases se observa que el comportamiento de BCE y fundamentalmente el de CCE es más homogéneo que el de *Bagging* y *ECOC*. Así, al considerar la clase que presenta mayor dificultad para ser aprendida (clase 6) se puede comprobar que los resultados logrados con CCE son mucho mejores que los conseguidos con los modelos de referencia (77,15% frente a al 68,15% obtenido por *Bagging* y el 72,33% dado por *ECOC*).

Al analizar la robustez frente al ruido se comprueba que, con un nivel de ruido inducido del 10% CCE es estadísticamente superior al resto de modelos estudiados.

Finalmente, al estudiar la relación entre la diversidad de los clasificadores base y la mejora del conjunto (Precisión del conjunto – Precisión media de los clasificadores base) se comprueba que, en general, la relación existente entre ambos parámetros es mucho más fuerte en CCE que en el resto de modelos analizados.

## 5.7. Resumen y Discusión

En este capítulo se ha analizado el comportamiento, en términos de diversidad y precisión de las arquitecturas BCE y CCE sobre dominios con discrepancias en: el número de clases; el número de atributos; el número de ejemplos de entrenamiento; la proporción de instancias pertenecientes a cada clase; el número de clases asignadas a cada ejemplo.

Los resultados experimentales obtenidos reflejan que ambas propuestas se pueden considerar de *carácter general*, pues, en esencia, resuelven correctamente todos los problemas sobre los que han sido evaluados. La única excepción detectada se refiere a la aplicación de CCE sobre un dominio *multietiqueta*. En este caso, la experimentación realizada (sección 5.5) pone de manifiesto que para aplicar este modelo a problemas en los que un ejemplo puede pertenecer de forma simultánea a dos o más clases se requieren mecanismos adicionales con los que discernir la pertenencia o no de un ejemplo a cada una de las clases.

Teniendo presente esta excepción, en la Tabla 5.49, se recoge el resumen de los dominios analizados, las principales características de los modelos construidos (número de neuronas ocultas y ciclos de entrenamiento) y las conclusiones derivadas del estudio comparativo referido a la precisión global de BCE y CCE sobre cada uno de ellos. Señalar, que en todos los casos expuestos, el número de módulos base (tanto de BCE como de CCE) coincide con el número de clases.

**Tabla 5.49. Resumen de los dominios estudiados, principales características de la arquitectura de BCE y CCE y resultados de la comparación estadística realizada. El símbolo (+)/(-) indica que, con un nivel de significancia del 0,05 y según el test estadístico aplicado BCE es mejor/peor que CCE. El símbolo (.) indica que los sistemas son estadísticamente equivalentes.**

Dominio	Ejemplos Entrenamiento	Ejemplos Test	Clases	Atributos	Desbalanceo	Neuronas ocultas	Ciclos	Test Estadístico	BCE vs CCE
SYNTHETIC	300	300	6	60	No	15	300	T-Test	(+)
SEMEION	796	796	10	256	No	20	300	T-Test	(+)
SEMEION MODIFICADO	620	620	10	256	Sí	20	300	T-Test	(+)
MNIST	60000	10000	10	784	No	50	500	McNemar	(-)
ASISTENTUR	450	450	9	1024	No	30	2000	F-Test	(.)
ASISTENTUR CON RUDIO	450	450	9	1024	No	30	2000	F-Test	(.)

Observando los valores recogidos en la Tabla 5.49 se puede comprobar que sobre dominios con pocos atributos la arquitectura BCE es más precisa que CCE. Por el contrario, sobre dominios con un gran número de atributos (MNIST y ASISTENTUR), el comportamiento de CCE es equivalente al de BCE o incluso mejor. Recaltar que es en estos últimos casos cuando realmente se explota el potencial de ambas arquitecturas, pues cuando el número de atributos es muy elevado, BCE y CCE son, en términos de

precisión, equivalentes a otras propuestas muy valoradas en la Bibliografía (Bagging y ECOC y OAO) pero más eficientes que éstas.

Analizando en términos generales la diversidad y la precisión de los módulos base que integran cada una de las arquitecturas propuestas, se comprueba que, salvo en SYNTHETIC, en el resto de dominios analizados el comportamiento de BCE y CCE se ajusta al esperado. Es decir, los módulos base de BCE son más precisos pero menos diversos que los de CCE. Además, la precisión de estos conjuntos es superior a la de cualquiera de sus miembros.

## Capítulo 6

### Conclusiones y Trabajos Futuros

Una de las áreas de investigación que durante las últimas décadas ha recibido mayor atención dentro del *Aprendizaje Automático* ha sido el diseño de *Conjuntos de Clasificadores*. Sin embargo, y a pesar del gran número de estudios realizados, la construcción de un conjunto de clasificadores óptimo en todas las situaciones es una meta que, por ahora, sigue siendo una utopía.

Muchos de los *Conjuntos de Clasificadores* propuestos hasta la fecha, han sido diseñados y, fundamentalmente, evaluados sobre dominios relativamente sencillos. Es decir, dominios en los que los ejemplos están descritos por un reducido número de atributos. Además, y dado que muchos de los algoritmos en los que se sustentan las tareas de clasificación fueron originalmente diseñados para resolver problemas binarios, un gran número de estas propuestas solo se pueden aplicar en dominios en los que únicamente hay dos categorías. Esto obliga a que para resolver un problema *multiclase* sea imprescindible descomponer el problema original en subproblemas dicotómicos.

En este trabajo de Tesis, se han presentado dos nuevas arquitecturas de conjuntos de clasificadores, *Binary-Complementary Ensemble* (BCE) y *Complementary-Complementary Ensemble* (CCE), diseñadas explícitamente para la resolución de problemas *multiclase* en los que los ejemplos están descritos por un elevado número de atributos. A nivel de clasificador base, estas arquitecturas se pueden considerar complementarias entre sí, pues mientras que en BCE se prima la precisión frente a la diversidad, en CCE se prima la diversidad frente a la precisión. Por tanto, ante determinadas particularidades del dominio dispondremos de la opción de decidir cuál de ellas resulta más adecuada.

Con el propósito de validar los métodos propuestos se ha realizado un exhaustivo análisis experimental. En este estudio los valores de precisión y diversidad obtenidos por nuestros modelos se han comparado con los obtenidos por otros métodos referenciados en la Bibliografía.

## 6.1. Conclusiones

En este epígrafe se indican las principales conclusiones extraídas al analizar los resultados empíricos derivados de la aplicación de BCE y CCE.

### **Diversidad**

El análisis de diversidad realizado sobre varios de los dominios estudiados pone de manifiesto que, tal y como era nuestro objetivo, los módulos base que integran CCE son bastante más diversos que los que componen BCE. Además, al establecer en términos estadísticos la relación entre la mejora del conjunto ( $P_{\text{CJTO}} - P_{\text{MÁX}}$ ) y las distintas medidas de diversidad se observa que la relación entre ambos parámetros es más acusada en CCE que en BCE. Es decir, en un mayor número de casos, la mejora en la precisión de CCE se puede justificar en términos de la diversidad inducida en sus módulos base. No obstante, y aunque tanto en BCE como en CCE la diversidad entre módulos se consigue fusionando varias de las técnicas propuestas para tal fin, los valores obtenidos por CCE, y por consiguiente por BCE, están muy alejados de los valores teóricos deseados. A pesar de ello, en los casos analizados, los valores de diversidad obtenidos por ambos modelos, y en especial por CCE, son muy superiores a los alcanzados con *Bagging*.

### **Precisión**

El análisis de precisión realizado corrobora que los módulos de BCE son, salvo alguna excepción, mucho más precisos que los de CCE. Es más, en ciertas ocasiones se ha podido observar que la precisión de alguno de los módulos de BCE es incluso mayor que la del propio conjunto. Esta circunstancia, pone de manifiesto que en ciertos dominios, en los que los ejemplos son claramente diferenciables y en los que el número de ejemplos conflictivos es muy reducido (caso de SYNTHEIC), el uso de los conjuntos de clasificadores no está plenamente justificado. Es decir, aunque en general, los conjuntos de clasificadores son mejor solución que los clasificadores simples, en ocasiones éstos últimos pueden ofrecer una solución igual o mejor que la lograda por un conjunto.

### **Desbalanceo de clases**

Al evaluar el comportamiento de BCE y CCE sobre un dominio en el que ciertas clases están claramente desbalanceadas se ha podido comprobar que ambos sistemas son buenas alternativas para este tipo de problemas. Aunque sobre el dominio estudiado se puede concluir que de forma genérica el comportamiento de ambos modelos es equivalente, es destacable que, al analizar el comportamiento por clases éste es complementario. Es decir, al analizar el comportamiento de BCE y CCE sobre las dos clases minoritarias más conflictivas, se da la circunstancia de que uno supera al otro en una clase y el otro supera al uno en la otra clase. Esta circunstancia se puede interpretar como un indicativo de la complementariedad existente entre ambas arquitecturas.

### Relación con el número de atributos

Los resultados experimentales obtenidos al evaluar las arquitecturas propuestas sobre las bases de datos SYNTHETIC y SEMEION parecen indicar que en dominios *multiclase* en los que el número de atributos no es excesivamente elevado BCE es mejor alternativa que CCE. Por el contrario, los resultados obtenidos sobre MNIST y ASISTENTUR sugieren que en dominios en los que el número de atributos es muy elevado, CCE es mejor alternativa que BCE<sup>16</sup>. Aunque en términos estadísticos algunas de estas conclusiones no se han podido justificar lo que es indudable es que en este último tipo de dominios y considerando simultáneamente eficiencia y eficacia, las arquitecturas propuestas son superiores a la red de neuronas simple, OAA, *Bagging* y ECOC. No obstante, al fusionar estos últimos modelos con un método de selección de características, la supremacía de CCE sólo se mantiene sobre MNIST. Por el contrario, los resultados experimentales obtenidos sobre ASISTENTUR muestran que, a pesar de que CCE es el modelo que presenta una menor tasa de error, su comportamiento es estadísticamente equivalente al de *Bagging*, ECOC y BCE. Sin embargo, al analizar la robustez frente al ruido de estos sistemas sobre ASISTENTUR se comprueba que, con un nivel de ruido inducido del 10%, CCE es superior al resto de modelos estudiados.

### Dominios jerárquicos

Al evaluar BCE sobre la base de datos Reuters se ha podido comprobar que BCE es perfectamente aplicable a dominios jerárquicos en los que los ejemplos pueden pertenecer, de forma simultánea, a más de una clase. Por el contrario, el estudio realizado con CCE pone de manifiesto que para poder aplicar este modelo a un problema *multietiqueta* es necesario contar con mecanismos adicionales para establecer el umbral con el que discernir si un ejemplo pertenece o no a una determinada clase. Al comparar estadísticamente los resultados de BCE con los obtenidos por Lewis [Lewis, et al., 2004] al usar tres de los métodos más valorados en el ámbito de la clasificación de textos (*svm*, *k-NN* y *Rocchio*), se concluye que si bien BCE es estadísticamente peor que la *máquina de soporte vectorial*, es estadísticamente superior al sistema basado en el algoritmo *Rocchio* y al sistema basado en *k-NN*.

### Esquema de integración Binario-Complementario

Finalmente señalar que, contrariamente a lo esperado, el análisis experimental realizado con BCE ha mostrado que los mejores valores, en lo que a simplicidad y precisión se refiere, se obtienen cuando la arquitectura de *integración binario-complementario* sigue un esquema en paralelo.

---

<sup>16</sup> Aunque en ASISTENTUR los resultados de CCE son mejores que los de BCE, desde el punto de vista estadístico ambos sistemas son equivalentes.

## 6.2. Trabajos Futuros

El trabajo desarrollado en esta Tesis plantea líneas de investigación que pueden ser estudiadas y desarrolladas en un futuro. Entre estas líneas se proponen las siguientes:

- Analizar y estudiar el comportamiento de BCE y CCE cuando los clasificadores que integran un módulo se construyen usando otros algoritmos de aprendizaje (por ejemplo, árboles de decisión, reglas de decisión o métodos estadísticos). Este estudio incluiría no sólo probar con otra técnica de aprendizaje sino, también, la construcción de conjuntos heterogéneos.
- Otro factor a tener en cuenta en futuros trabajos es la influencia que, en ambas arquitecturas, tiene la combinación de distintos métodos de selección de características.
- Uno de los factores que pueden mejorar la precisión de CCE parece estar relacionado con la construcción y selección de los módulos base que lo integran. Por ello, resulta interesante estudiar las ventajas que puede aportar el uso de técnicas evolutivas o técnicas de aprendizaje no supervisado como herramientas para determinar la matriz de códigos de CCE.
- Las dificultades encontradas a la hora de aplicar CCE a un dominio *multietiqueta* sugieren que es necesario proponer técnicas eficientes que permitan establecer la pertenencia o no de un ejemplo a una determinada clase. Aunque este estudio resulta imprescindible en el caso de CCE creemos que su aplicación a BCE permitirá mejorar los resultados obtenidos con este último modelo.
- Los resultados experimentales parecen indicar que el potencial de CCE aumenta en relación con el número de clases y el número de atributos. Dado que esta afirmación no va más allá de un simple conjetura, como línea de investigación inmediata, se plantea analizar de forma más exhaustiva la influencia que estos factores tienen en el comportamiento de CCE. Es decir, se plantea la necesidad de encontrar nuevos indicios con los que corroborar o refutar esta hipótesis.



## Bibliografía

- [Addis et al., 2008] Addis, A., Armano, G. and Vargiu, E. (2008): A Progressive Filtering Approach to Hierarchical Text Categorization. *Communications of SIWN* Vol.5, pp. 28-32.
- [Allwein et al, 2000] Allwein, E.L., Schapier, R. E. and Singer, Y. (2000): Reducing Multiclass to Binary: A Unifying Approach for Margin Classifiers. *Journal of Machine Learning*, Vol. 1, pp 113-141
- [Alpaydin, 1999] Alpaydin, E. (1999): Combined 5x2cv F-Test for Comparing Supervised Classification Learning Algorithms. *Neural Computation* Vol. 11, pp. 1885-1892.
- [Álvarez, 2007] Álvarez, R. (2007): *Estadística Aplicada a las Ciencias de la Salud*. Ediciones Díaz de Santos.
- [Anand, et al., 1995] Anand, R., Mehrotra, K., Chilukuri, K. M. and Rank S. (1995): Efficient Classification for Multiclass Problems Using Modular Neural Networks. *Transactions on Neural Networks*, Vol. 6, Nº 1, pp 117-124
- [Bahler & Navarro, 2000] Bahler, D. and Navarro L. (2000): Methods for Combining Heterogeneous Sets of Classifiers. *17th National Conference on Artificial Intelligence (AAAI)*, Workshop on New Research Problems for Machine Learning
- [Blum & Langley, 1997] Blum, A. L. and Langley, P. (1997): Selection of Relevant Features and Examples in Machine Learning. *Artificial Intelligence*, Vol. 97, pp. 245-271.
- [Breiman, 1996] Breiman, L. (1996): Bagging Predictors. *Machine Learning*, Vol. 24, Nº 2, pp. 123-140.
- [Bryll et al., 2003] Bryll, F., Gutierrez-Osuna, R. and Quek, F. (2003): Attribute Bagging : Improving Accuracy of Classifier Ensembles by Using Random Feature Subsets. *Pattern Recognition*, Vol. 36, Nº 6, pp. 1291-1302
- [Burns & Danyluk, 2000] Burns, B.D. and Danyluk A.P. (2000): Feature selection vs. Theory Reformulation: A Study of Genetic Refinement of Knowledge-Based Neural Networks. *Machine Learning*, Vol 38, Nº 1-2, pp. 89-107.
- [Castillo & Serrano, 2004] Castillo, M.D. del and Serrano, J. I. (2004): A Multistrategy Approach for Digital Text Categorization from Imbalanced Documents. *ACM SIGKDD Explorations Newsletter - Special issue on learning from imbalanced datasets*. Vol. 6, Nº 1
- [Curteanu, et al., 2010] Curteanu, S., Leon, F., Furtuna, R., Dragoi, E. N. and Curteanu, N. (2010): Comparison between Different Methods for Developing Neural Network Topology Applied to a Complex Polymerization Process. *IEEE World Congress on Computational Intelligence*, pp. 18-23.
- [Dietterich, 1997] Dietterich, T. G. (1997): Machine Learning Research: Four Current Directions. *AI Magazine*. Vol. 18, Nº 4, pp. 97-136

- [**Dietterich, 2000a**] Dietterich, T. G. (2000): Ensemble Methods in Machine Learning. Proceedings of the First International Workshop on Multiple Classifier Systems. Lecture Notes in Computer Science; Vol. 1857, pp. 1-15
- [**Dietterich, 2000b**] Dietterich, T. G. (2000): An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization. *Machine Learning*, Vol 40, No. 2, p.p. 139-157
- [**Dietterich & Bakiri, 1995**] Dietterich, T. G., and Bakiri, G. (1995): Solving Multiclass Learning Problems via Error-Correcting Output Codes. *Journal of Artificial Intelligence Research*, Vol. 2, pp 263-286.
- [**Duin & Tax, 2000**] Duin, R. and Tax, D. (2000): Experiments with Classifier Combining Rules. *Proceedings of the first International Workshop on Multiple Classifier Systems*. Lecture Notes in Computer Science, Vol. 1857, pp. 16-29.
- [**Esuli, et al., 2008**] Esuli, A., Fagni, T. and Sebastiani, F. (2008): Boosting Multi-label Hierarchical Text Categorization. *Information Retrieval*, Vol. 11, pp. 287-313.
- [**Everitt, 1977**] Everitt, B. S. *The analysis of contingency tables* (1977): Chapman and Hall, London.
- [**Forman, 2004**] Forman, G. (2004): A Pitfall and Solution in Multi-Class Feature Selection for Text Classification. *Proceedings of the 21st International Conference on Machine Learning*
- [**Frank & Asuncion, 2010**] Frank, A. and Asuncion, A. (2010): UCI Machine Learning Repository [<http://archive.ics.uci.edu/ml>]. Irvine, CA: University of California, School of Information and Computer Science
- [**Frank & Witten, 1998**] Frank, E. and Witten, I. (1998): Generating Accurate Rule Sets Without Global Optimization. *Proceedings of the Fifteenth International Conference on Machine Learning*, pp 144–151. Morgan Kaufmann.
- [**Freund & Schapire, 1997**] Freund Y. and Schapire, R. E. (1997): A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, Vol. 55, N° 1, pp. 119-139.
- [**Friedman, 1996**] Friedman, J. (1996): *Another approach to polychotomous classification*. Technical report, Stanford University.
- [**Fu et al., 2005**] Fu, Q., Hu, S. and Zhao, S. (2005): Clustering-based selective neural network ensemble. *Journal of Zhejiang University SCIENCE*, Vol. 6A, N° 5, pp. 387-392
- [**Galar et al., 2011**] Galar, M., Fernández, A., Barrenechea, E., Bustince, H., Herrera, F. (2011): An Overview of Ensemble Methods for Binary Classifiers in Multi-class Problems: Experimental Study on One-vs-One and One-vs-All Schemes. *Pattern Recognition*, Vol 44, N° 8, pp. 1761-1776
- [**Galindo et al., 2000**] Galindo, P.L., Pizarro-Junquera, J., and Guerrero, E. (2000): Multiple Comparison Procedures for Determining the Optimal Complexity of a Model. Advances. *Pattern Recognition. Lecture Notes in Computer Science*, Vol. 1876/2000, p.p 796-805.

- [**García-Pedrajas, et al., 2007**] García-Pedrajas, N. Gacía-Osorio, C. and Fyfe, C. (2007): Nonlinear Boosting projections for Ensemble Construction. *Journal of Machine Learning Research* Vol. 8, p.p. 1-33.
- [**García-Pedrajas & Haro, 2008**] García-Pedrajas, N. and Haro García, A. (2008): Output Coding Methods: Review and Experimental Comparison. *Pattern Recognition Techniques, Technology and Applications*. Cap. 14. Peng-Yeng Yin (ed). ISBN 978-953-7619-24-4. I-Tech, Vienna, Austria
- [**García-Pedrajas & Ortiz-Boyer, 2006**] García-Pedrajas, N., Ortiz-Boyer, D. (2003): Improving Multiclass Pattern Recognition by the Combination of Two Strategies. *Transactions on Pattern Analysis and Machine Intelligence*, Vol. 28, No. 6, pp 1001-1006.
- [**García-Pedrajas & Ortiz-Boyer, 2011**] García-Pedrajas, N., Ortiz-Boyer, D. (2011): An Empirical Study of Binary Classifier Fusion Methods for Multiclass Classification. *Information Fusion*, Vol. 12, N° 2, pp 111-130
- [**Gasmi & Merouani, 2005**] Gasmi I. and Merouani H. (2005): Towrads a Method of Automatic Design of Multi\_Classifiers System Based Combination. *Word Academy of Science, Engineering and Technology*, No 6, p.p 82-87
- [**Giacinto, 1998**] Giacinto, G. Design of Multiple Classifier Systems (1998): En <http://www.fizyka.umk.pl/ftp/incoming/duch/Komitety/Multiple-classif-Giacinto-PhD.pdf>
- [**Goebel & Yan, 2004**] Goebel, K. and Yan W. (2004): Choosing Classifiers for Decision Fusion. *Proceedings of the Seventh International Conference on Information Fusion*, Vol. 1, pp. 563-568
- [**Goldberg , 1989**] Goldberg, D.E. (1989): *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley ed.
- [**Gutiérrez et al., 2005**] Gutiérrez, G., Sanchis, A., Isasi, P., Molina J.M. and Galván I. M. (2005): Non-direct Encoding Method Based on Cellular Automata to Design Neural Network Architectures. *Computing and Informatics*. Vol. 24, pp. 225-247.
- [**Hall, 1999**] Hall, M.A. (1999): *Correlation-based Feature Selection for Machine Learning*. Ph.D diss. Hamilton, NZ: Waikato University. Department of Computer Science.
- [**Hansen & Salamon, 1990**] Hansen, L., and Salamon, P. (1990): Neural Network Ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol, 12, No 19, pp. 993-1001.
- [**Hao et al., 2003**] Hao, H. , Liu, C. and Sako, H. (2003): Comparison of Genetic Algorithm and Sequential Search Methods for Classifier Subset Selection. *Proceedings of the Seventh International Conference on Document Analysis and Recognition*, p.p. 765- 770
- [**Hastie & Tibshirani, 1998**] Hastie, T. and Tibshirani, R. (1998): Classification by Pairwise Coupling. *The Annals of Statistics*, Vol. 26, No. 2, pp 451-471.

- [Ho, 1998] Ho, T. K. (1998): The Random Subspace Method for Constructing Decision Forest. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol, 20, N° 8, pp. 832-844.
- [Jain et al., 2000] Jain, A. K., Duin, R. P. W. and Jianchang Mao. (2000): Statistical Pattern Recognition: a Review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, N°1, pp. 4-37.
- [Joachims, 1998] Joachims, T. (1998): Text Categorization with Support Vector Machines: Learning with many Relevant Features. *European Conference of Machine Learning*, pp. 137-142.
- [Kim et al., 2006] Kim Y., Street W. N. and W., Menczer, F. (2006): Optimal ensemble construction via meta-evolutionary ensembles. *Expert Systems with Applications*, Vol. 30, No. 4, pp 705-714.
- [Kolen & Pollack, 1990] Kolen, J.F. and Pollack, J.B. (1990): Back-Propagation is Sensitive to Initial Conditions. *Complex Systems*, Vol. 4, No. 3, pp 269-280.
- [Kraipeerapun et al., 2006] Kraipeerapun, P., Fung, C.C. and Wong, K.W. (2006): Multiclass Classification using Neural Networks and Interval Neutrosophic Sets. *5th WSEAS International Conference on Computational Intelligence, Man-Machine Systems and Cybernetics*, 20-22 November 2006, Venice, Italy.
- [Kuncheva, 2002] Kuncheva, L.I. (2002): Switching between Selection and Fusion in Combining Classifiers: An experiment. *IEEE Transactions on SMC*, Vol. 32, No. 2, p.p 146-156.
- [Kuncheva, 2005] Kuncheva, L.I. (2005): *Combining Pattern Classifiers, Methods and Algorithms*. New York, NY: Wiley Interscience.
- [Kuncheva & Whitaker, 2001] Kuncheva L.I. and Whitaker, C.J. (2001): Ten Measures of Diversity in Classifier Ensembles: Limits for two Classifiers. *IEEE Workshop on Intelligent Sensor Processing*.
- [Kuncheva & Whitaker, 2003] Kuncheva L.I. and Whitaker, C.J. (2003): Measures of diversity in classifier ensembles, *Machine Learning* , N° 51, p.p. 181-207.
- [Lazarevic & Obradovic, 2001] Lazarevic, A. and Obradovic Z. (2001): Effective Pruning of Neural Network Classifier Ensembles. *Proceedings. IJCNN '01. International Joint Conference on Neural Networks*, Vol.2, pp.796-801
- [LeCun et al., 1995] LeCun Y., Jackel, L. D., Bottou, L., Cortes C., Denker, J. S., Drucker, H., Guyon, I., Muller, U. A., Sackinger, E., Simard, P., and Vapnik V. (1995): Learning Algorithms For Classification: A Comparison On Handwritten Digit Recognition. *Neural Networks: The Statistical Mechanics Perspective*, pp. 261-276. Ed. J. H. Oh, C. Kwon, and S. Cho, World Scientific.
- [Ledezma, 2004] Ledezma, A. I. (2004): *Aprendizaje Automático en Conjuntos de Clasificadores Heterogéneos y Modelado de Agentes*. UC3M. Tesis Doctoral.
- [Ledezma et al., 2010] Ledezma, A., Aler, A., Sanchis A. and Borrajo, D. (2010): GA-stacking: Evolutionary stacked generalization. *Intelligent Data Analysis* Vol 14, No. 1, pp 89-119.

- [**Lewis, 2004**] Lewis, D. D. (2004): *RCV1-v2/LYRL2004: The LYRL2004 Distribution of the RCV1-v2 Text Categorization Test Collection* (12-Apr-2004 Version). [http://www.jmlr.org/papers/volume5/lewis04a/lyrl2004\\_rcv1v2\\_README.htm](http://www.jmlr.org/papers/volume5/lewis04a/lyrl2004_rcv1v2_README.htm).
- [**Lewis et al., 2004**] Lewis, D. D., Yang, Y., Roe, T. G. and Li F. (2004): RCV1: A New Benchmark Collection for Text Categorization Research. *Journal of Machine Learning Research*, Vol 5., pp. 361-397
- [**Lézoray & Cardot, 2008**] Lézoray, O. and Cardot, H. (2008): Comparing Combination Rules of Pairwise Neural Networks Classifiers. *Neural Processing Letters*, Vol 27, Nº 1, pp. 43-56.
- [**Liu & You, 2002**] Liu, H. and Yu L. (2002): *Feature selection for data mining*, Research Technical Report. Available in: <http://www.public.asu.edu/~huanliu/surfs02.ps>
- [**Martin, 1995**] Martin, B. (1995): *Instance-based learning : Nearest neighbor with generalization*. Master's thesis, University of Waikato.
- [**Martínez-Muñoz, 2006**] Martínez-Muñoz, G. (2006): Clasificación mediante conjuntos. Tesis Doctoral. Dpto. de Ingeniería Informática. Universidad Autónoma de Madrid. <http://arantxa.ii.uam.es/~gonzalo/publications/tesis.pdf>
- [**Masulli & Valentini, 2000**] Masulli, F. and Valentini, G. (2000): Comparing decomposition methods for classification. *Proceedings Fourth International Conference on Knowledge-Based Intelligent Engineering Systems and Allied Technologies*, Vol. 2, pp. 188-791.
- [**Michalski, 1983**] Michalski, R. S. (1983): A Theory and Methodology of Inductive Learning. *Machine Learning: An Artificial Intelligence Approach*, R. S. Michalski, T. J. Carbonell and T. M. Mitchell (Eds.), pp. 83-134, TIOGA Publishing Co., Palo Alto.
- [**Michalski, 1994**] Michalski, R. S., (1994): Inferential Theory of Learning: Developing Foundations for Multistrategy Learning. *Machine Learning: A Multistrategy Approach, Vol. IV*, San Mateo, CA, pp. 3-61, Morgan Kaufmann.
- [**Minsky, 1954**] Minsky, M. L. (1954): *Neural nets and the brain-model problem*. Ph.D. dissertation, Universidad de Princeton.
- [**Mitchell, 1997**] Mitchell, T. M. (1997): *Machine Learning*. McGraw-Hill.
- [**Molina & García, 2006**]. Molina, J.M., García, J. (2006): *Técnicas de Análisis de Datos. Aplicaciones Prácticas utilizando Microsoft Excel y Weka*. <http://www.giaa.inf.uc3m.es/docencia/II/ADatos/apuntesAD.pdf>
- [**Mooney & Ourston, 1994**] Mooney, R.J. and Ourston, D. (1994): A Multistrategy Approach to Theory Refinement. *Machine Learning: A Multistrategy Approach, Vol. 4*, Michalski, R.S. and Tecuci, G. (eds.). Morgan Kaufmann Publishers, pp. 141-164.
- [**Murphey et al., 2007**] Murphey, Y. L., Wang, H., Ou, G. and Feldkamp, L.A. (2007): OAHO: an Effective Algorithm for Multi-Class Learning from Imbalanced Data. *Proceedings of International Joint Conference on Neural Networks*.

- [**Muthuramalingam et al., 2007**] Muthuramalingam, A., Himavathi, S., and Srinivasan, E. (2007): Neural network implementation using FPGA: Issues and application. *International Journal of Information Technology*, Vol. 4, N° 2, pp. 86-92.
- [**Oliveira et al., 2003**] Oliveira, L.S., Sabourin, R., Bortolozzi, F. and Suen, C.Y. (2003): A Methodology for Feature Selection Using Multi-objective Genetic Algorithms for Handwritten Digit String Recognition. *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI)*, Vol. 17, n° 6, pp. 903-929.
- [**Oong & Mat Isa, 2012**] Oong, T.H, Mat Isa, N.A. (2012): One-against-all Ensemble for Multiclass Pattern Classification. *Applied Soft Computing*. doi:10.1016/j.asoc.2011.12.004.
- [**Opitz, 1999**] Opitz, D.W. (1999): Feature selection for ensembles. *Proceedings of the 16<sup>th</sup> International Conference on Artificial Intelligence*, pp. 379–384.
- [**Opitz & Maclin, 1999**] Opitz D. and Maclin, R. (1999): Popular Ensemble Methods: An Empirical Study. *Journal of Artificial Intelligence Research*, Vol.11, pp.169-198.
- [**Ordóñez et al., 2008**] Ordóñez, F.J., Ledezma, A. and Sanchis, A. (2008): Genetic Approach for Optimizing Ensembles of Classifiers. *Proceedings of the Twenty-First International Florida Artificial Intelligence Research Society Conference*, pp. 89-94.
- [**Ormondi & Rajapakse, 2006**] Ormondi, A.R., Rajapakse, J.C (2006): *FPGA Implementations of Neural Networks*. Springer, ISBN 0-387-38485-0
- [**Ou & Murphey, 2007**] Ou, G. and Murphey, L (2007): Multi-class pattern classification using neural networks. *Pattern Recognition*, Vol. 40 n° 1, pp. 4-18.
- [**Parmanto et al., 1996**] Parmanto, B., Munro, P. and Doyle, H. (1996): Improving Committee Diagnosis with Resampling Techniques. *Advances in Neural Information Processing Systems*, Vol. 8, pp. 882–888.
- [**Poddar & Rao, 1993**] Poddar, P. and Rao, P.V.S. (1993): Hierarchical Ensemble of Neural Networks. *IEEE International Conference on Neural Networks*, Vol.1, pp. 287-292.
- [**Polikar, 2006**] Polikar, R. (2006): Ensemble Based Systems in Decision Making. In *Circuits and Systems Magazine, IEEE*, Vol. 6, N° 3, pp. 21-45.
- [**Polikar, 2012**] Polikar, R. (2012): Ensemble Learning. In *Ensemble Machine Learning. Methods and Applications*, Zang, Cha and Ma, Yunquian (Eds). Springer Science and Business Media, pp. 1-33.
- [**Pozo, 1989**] Pozo, J. I. (1989): *Teorías Cognitivas del Aprendizaje*. Ediciones Morata. Madrid.
- [**Quinlan, 1986**] Quinlan, J. R. (1986): Induction of Decision Trees. *Machine Learning*, Vol. 1, N° 1, pp. 81-106

- 
- [Quinlan, 1993] Quinlan, J.R. (1993): *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo
- [Quinlan, 1996] Quinlan, J. R. (1996): Bagging, Boosting, and C4.5. *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, Portland, OR.
- [de Raedt & Bruynooghe, 1994] de Raedt, L. and Bruynooghe, M. (1994): Interactive Theory Revision. *Machine Learning: A Multistrategy Approach*, Vol. 4, Michalski, R.S. and Tecuci, G. (eds.). Morgan Kaufmann Publishers, pp. 239-263.
- [Ranawana, 2006] Ranawana, R. (2006): Multi-Classifer Systems- A review and Roadmap for Developers. *International Journal of Hybrid Intelligent Systems*, Vol. 3, N° 1, pp 35-61.
- [Rifkin & Klautau, 2004] Rifkin R. and Klautau A. (2004): In defence of one-vs-all classification. *Journal of Machine Learning Research*, Vol. 5, pp. 101-141.
- [Rijsbergen, 1979] Rijsbergen, C. J. van (1979): *Information Retrieval*. Butterworth, 1979. ISBN 0-408-70929-4.
- [Rocchio, 1971] Rocchio, J. J. (1971): Relevance Feedback in Information Retrieval. *The SMART Retrieval System: Experiments in Automatic Document Processing*, pp 313-323. Prentice-Hall.
- [Rokach, 2010] Rokach, L. (2010): Ensemble-based classifiers. *Artificial Intelligence Review*, Vol. 33, N° 1-2, pp 1-39.
- [Roli et al., 2001] Roli, F., Giacinto, G., and Vernazza, G. (2001): Methods for Designing Multiple Classifier Systems. *Lecture Notes in Computer Science*, Vol. 2096, pp. 78-87.
- [Rose et al., 2002] Rose, T., Stevenson, M. and Whitehead, M. (2002): The Reuters Corpus Volume 1- from Yesterday's News to Tomorrow's Language Resources. *In Proceedings of the Third International Conference on Language Resources and Evaluation*.
- [Rumelhart & McClelland, 1988] Rumelhart, D. E., and McClelland, J.L (1988): *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. Cambridge, MA: MIT Press.
- [Rumelhart, et al., 1986] Rumelhart, D.E., Hinton, G.E., and Williams, R.J. (1986): Learning Internal Representations by Error Propagation. In D.E. Rumelhart and J.L. McClelland (Eds) *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Vol 1.
- [Russell & Norvig, 2003] Russell, S.J. and Norvig, P. (2003): *Artificial Intelligence: A Modern Approach*. Prentice Hall.
- [Saitta, 2001] Saitta, L. (2001): Integrated Architectures for Machine Learning. *Machine Learning and its Applications. Lecture Notes in Computer Science*, Vol 2046, pp. 218-229.
- [Sansone, et al., 2011] Sansone, C., Kittler, J. and Roli, F. (Eds). *Proceedings of International Workshop on Multiple Classifier Systems*. Lecture Notes in Computer Science. Springer (2000-2011).
-

- [Schapire, 1990] Schapire, R. E. (1990): The Strength of Weak Learnability. *Machine Learning*, Vol. 5, N° 2, pp. 197-227.
- [Schapire, 1997] Schapire, R. E. (1997): Using Output Codes to Boost Multiclass Learning Problems. *Proceedings of the Fourteenth International Conference of Machine Learning*.
- [Sharkey, 1999] Sharkey, A.J.C. (1999): *Multi-Net Systems. Combining Artificial Neural Nets*. Amanda J.C. Sharkey ed., Springer-Verlag New York pp 1-29
- [Sharkey & Sharkey, 1997] Sharkey, A.J.C and Sharkey N.E., (1997): Combining Diverse Neural Net. *The Knowledge Engineering Review*, Vol. 12, N° 3, pp 231-247.
- [Sharkey et al., 2000] Sharkey, A. J. C., Sharkey, N. E., Gerecke, U. and Chandroth, G. O. (2000): The “Test and Select” Approach to Ensemble Combination. *Lecture Notes in computer Science*, Vol. 1857, pp. 30-44.
- [Sheskin 2000] Sheskin, D. J. (2000): *Handbook of parametric and nonparametric statistical procedures*. Second Edition. Chapman & Hall/CRC
- [Sesmero et al., 2007a] Sesmero, M.P, Alonso-Weber, J.M., Gutiérrez, G., Ledezma, A. and Sanchis, A. (2007): Testing Feature Selection in Traffic Signs. *In Proceedings of the 11<sup>th</sup> International Conference on Computer Aided Systems*, pp. 396–398.
- [Sesmero et al., 2007b] Sesmero, M.P, Alonso-Weber, J.M., Gutiérrez, G., Ledezma, and A., Sanchis, A. (2007): Specialized Ensemble of Classifiers for Traffic Sign Recognition. *In Computational and Ambient Intelligence. Lecture Notes in Computer Science*. Vol. 4507, pp. 733–740.
- [Sesmero, et al., 2012] ] Sesmero, M.P, Alonso-Weber, J.M., Gutiérrez, G., Ledezma, and A., Sanchis, A. (2012): A new artificial neural network ensemble based on feature selection and class recoding. *Neural Computing & Applications*, Vol, 21, N° 4, pp. 771-783.
- [Somerén, 2001] Someren, M. van (2001): Model Class Selection and Construction: Beyond the Procrustean Approach to Machine Learning Applications. *Machine Learning and its Applications. Lecture Notes in Computer Science*, Vol 2046, pp. 196-217.
- [Spiegel, 1976] Spiegel, R. M. (1976): *Teoría Y Problemas de Probabilidad y Estadística*. McGraw Hill.
- [Srinivas & Patnaik, 1994] Srinivas, M. and Patnaik, L.M. (1994): Genetic Algorithms: A survey, *Computer*, Vol. 27, N° 6, pp. 17-26.
- [Stathakis, 2009] Stathakis, D. (2009): How many hidden layers and nodes? *International Journal of Remote Sensing*, Vol. 30, N° 8, pp. 2133-2147.
- [Tax & Duin, 2002] Tax, D.M. J. and Duin R. P. W. (2002): Using Two-Class Classifiers for Multiclass Classification. *16th International Conference on Pattern Recognition* Vol. 2.



- [Tsymbal, et al., 2003] Tsymbal, A., Pechenizkiy, M. and Cunningham, P (2003): *Diversity in Ensemble Feature Selection*. Technical Report TCD-CS-2003-44, Computer Science Department Trinity College Dublin. <https://www.cs.tcd.ie/publications/tech-reports/reports.03/TCD-CS-2003-44.pdf>
- [Tsymbal, et al., 2006] Tsymbal, A., Pechenizky, M. and Cunningham, P. (2006): Diversity in Search Strategies for Ensemble Feature Selection. *Information Fusion*, Vol.6, N° 1, pp. 83-98
- [Whitaker and Kuncheva, 2003] Whitaker C.J. and Kuncheva, L.I. (2003): Examining the relationship between majority vote accuracy and diversity in bagging and boosting, *Technical Report, School of Informatics, University of Wales, Bangor*. Disponible en: [http://pages.bangor.ac.uk/~mas00a/papers/lkcw\\_tr.pdf](http://pages.bangor.ac.uk/~mas00a/papers/lkcw_tr.pdf)
- [Witten & Frank, 2005] Witten, I.H. and Frank, E. (2005): *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, San Francisco
- [Wolpert, 1992] Wolpert, D. (1992): Stacked Generalization. *Neural Networks*, Vol. 5, pp.241-259.
- [Woods et al., 1997] Woods, K., Kegelmeyer, W. P. and Browyer, K. (1997): Combination of Multiple Classifiers Using Local Accuracy Estimates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol 19, N° 4, pp. 405-410.
- [Yang, 1999] Yang, Y (1999): An Evaluation of Statistical Approaches to Text Categorization. *Information Retrieval*, 1(1):69–90.
- [Yan & Liu, 1999] Yang, Y. and Liu X. (1999): A Re-examination of Text Categorization Methods. *Proceedings of the 22<sup>nd</sup> ACM International Conference on Research and Development in Information Retrieval*, pp 42-49
- [Yan & Pedersen, 1997] Yang, Y. and Pedersen J. O. (1997): A Comparative Study on Feature Selection in Text Categorization. *Fourteenth International Conference on Machine Learning*, p.p. 412-420
- [Zhang, 2000] Zhang, G. (2000): Neural networks for classification: a survey. *IEEE Transactions on Systems, Man, and Cybernetics: Applications and Reviews*, Vol. 30, N° 4, pp. 451-462.
- [Zhang & Yunquian, 2012] Zhang, C. and Yunquian M. Editors. (2012): *Ensemble Machine Learning. Methods and Applications*. Springer
- [Zheng, 1996] Zheng, Z. (1996): Constructing New Attributes for Decision Tree Learning, Ph.D. Thesis, University of Sydney, <http://citeseer.ist.psu.edu/zheng96constructing.html>
- [Zhou & Tang, 2003]. Zhou, Z. and Tang, W. (2003): Selective Ensemble of Decision Trees. *Rough Sets, Fuzzy Sets, Data Mining, and Granular*. *Lecture Notes in Computer Science*, Vol. 2639.
- [Zhu et al., 2004] Zhu, X., Wu, X. and Yang, Y (2004): Dynamic Selection for effective mining from noisy data streams. *Proceedings of Fourth IEEE International Conference on Data Mining*, pp. 205-312



## Apéndice A

### Synthetic: $P_{MÁXIMA}$ vs $P_{BCE}$

En este apéndice se muestran algunos de los resultados obtenidos al evaluar BCE sobre la base de datos *Synthetic*.

**Tabla A.1. Synthetic.  $P_{MÁX}$ ,  $P_{MÍN}$ ,  $P_{MEDIA}$  y  $P_{BCE}$  para los 600 modelos construidos. En rojo se señalan los casos en los que  $P_{BCE} < P_{MÁX}$**

Partición	$P_{MÁXIMA}$	$P_{MÍNIMA}$	$P_{MEDIA}$	$P_{BCE}$	Partición	$P_{MÁXIMA}$	$P_{MÍNIMA}$	$P_{MEDIA}$	$P_{BCE}$
1-1	95,67%	79,00%	85,39%	96,67%	1-2	93,33%	79,33%	84,06%	96,00%
1-1	95,67%	77,00%	85,44%	96,33%	1-2	90,33%	78,33%	84,28%	95,67%
1-1	95,67%	77,00%	85,17%	96,33%	1-2	93,67%	78,67%	84,33%	96,00%
1-1	94,67%	77,33%	85,06%	96,67%	1-2	92,67%	79,00%	84,39%	95,00%
1-1	95,67%	77,67%	85,56%	96,33%	1-2	92,67%	77,67%	84,17%	96,00%
1-1	93,67%	78,67%	85,56%	96,67%	1-2	92,67%	78,33%	84,39%	96,00%
1-1	95,33%	77,67%	84,67%	95,67%	1-2	92,00%	77,67%	83,61%	95,33%
1-1	95,67%	77,67%	85,61%	97,00%	1-2	92,33%	77,67%	84,00%	96,00%
1-1	97,00%	76,00%	86,06%	96,33%	1-2	89,67%	78,67%	83,94%	96,33%
1-1	96,33%	79,33%	86,89%	96,67%	1-2	88,00%	78,67%	83,00%	95,33%
2-1	92,67%	77,67%	83,11%	95,00%	2-2	93,67%	79,33%	84,50%	98,33%
2-1	90,00%	78,33%	82,17%	93,67%	2-2	94,33%	81,00%	86,11%	98,67%
2-1	89,67%	77,67%	82,22%	94,33%	2-2	95,33%	80,67%	86,17%	98,33%
2-1	92,00%	78,00%	82,89%	95,00%	2-2	93,67%	81,33%	85,11%	98,67%
2-1	89,33%	78,67%	82,67%	95,00%	2-2	96,33%	79,33%	85,89%	98,67%
2-1	89,67%	78,33%	83,11%	96,00%	2-2	94,33%	80,00%	85,00%	98,33%
2-1	91,00%	77,67%	82,61%	95,00%	2-2	92,33%	80,33%	84,89%	99,00%
2-1	91,33%	78,00%	83,50%	97,00%	2-2	97,00%	80,67%	85,44%	98,33%
2-1	93,00%	77,67%	82,44%	93,67%	2-2	94,67%	79,67%	85,94%	99,00%
2-1	91,33%	77,67%	82,17%	96,00%	2-2	95,00%	79,33%	85,22%	98,67%

**Tabla A.1. Continuación.**

Partición	P <sub>MÁXIMA</sub>	P <sub>MÍNIMA</sub>	P <sub>MEDIA</sub>	P <sub>BCE</sub>	Partición	P <sub>MÁXIMA</sub>	P <sub>MÍNIMA</sub>	P <sub>MEDIA</sub>	P <sub>BCE</sub>
3-1	93,33%	77,33%	84,06%	96,67%	3-2	93,67%	76,67%	83,33%	97,67%
3-1	93,33%	80,00%	84,17%	96,33%	3-2	91,67%	80,00%	84,72%	98,67%
3-1	93,67%	78,67%	83,67%	96,00%	3-2	94,33%	79,67%	85,33%	99,33%
3-1	93,33%	78,33%	82,44%	96,67%	3-2	93,33%	80,00%	84,50%	98,33%
3-1	91,33%	79,00%	83,67%	96,67%	3-2	94,67%	79,67%	85,78%	98,33%
3-1	93,67%	79,00%	84,50%	97,00%	3-2	93,00%	79,67%	84,83%	99,00%
3-1	90,67%	78,67%	83,67%	97,00%	3-2	95,00%	79,00%	83,94%	98,67%
3-1	89,67%	79,00%	83,06%	96,33%	3-2	93,33%	79,33%	85,44%	98,67%
3-1	93,33%	78,00%	84,17%	96,33%	3-2	93,33%	79,67%	85,06%	98,67%
3-1	93,67%	78,67%	84,22%	96,33%	3-2	94,00%	79,33%	85,17%	98,33%
4-1	92,67%	79,00%	84,72%	99,33%	4-2	93,33%	76,67%	82,39%	93,33%
4-1	95,00%	80,33%	86,06%	99,67%	4-2	93,67%	76,67%	82,61%	94,00%
4-1	93,33%	80,67%	86,00%	99,00%	4-2	93,67%	76,33%	82,94%	95,67%
4-1	95,00%	79,33%	85,78%	99,33%	4-2	93,67%	75,33%	82,28%	93,67%
4-1	91,67%	79,67%	84,89%	99,67%	4-2	93,67%	75,33%	83,06%	94,67%
4-1	93,67%	80,00%	85,67%	99,00%	4-2	94,00%	76,00%	83,11%	94,00%
4-1	95,00%	79,67%	85,28%	99,67%	4-2	93,67%	76,67%	82,72%	94,00%
4-1	94,33%	80,67%	86,22%	99,67%	4-2	93,67%	75,33%	83,17%	94,33%
4-1	92,67%	80,67%	85,39%	99,33%	4-2	93,67%	76,33%	83,89%	96,00%
4-1	94,67%	80,00%	86,33%	100,0%	4-2	93,33%	76,67%	83,50%	94,00%
5-1	96,33%	76,67%	84,11%	98,00%	5-2	95,33%	78,00%	83,28%	97,33%
5-1	93,33%	79,00%	84,11%	96,33%	5-2	94,67%	78,00%	84,17%	96,00%
5-1	92,00%	78,67%	83,83%	97,33%	5-2	92,67%	78,67%	82,83%	97,67%
5-1	92,67%	78,33%	83,83%	97,00%	5-2	91,00%	79,00%	82,89%	97,00%
5-1	91,67%	79,00%	84,00%	97,00%	5-2	89,67%	78,67%	83,33%	97,33%
5-1	94,00%	79,00%	84,72%	97,33%	5-2	94,33%	77,67%	84,11%	96,67%
5-1	92,67%	78,67%	83,89%	97,67%	5-2	97,67%	78,00%	84,50%	96,67%
5-1	92,33%	78,00%	83,22%	98,00%	5-2	95,67%	79,00%	84,50%	97,67%
5-1	93,00%	73,33%	82,89%	97,67%	5-2	93,33%	78,33%	82,78%	96,33%
5-1	91,00%	79,00%	83,94%	97,33%	5-2	95,33%	78,67%	84,28%	96,33%
6-1	90,67%	79,00%	83,28%	96,00%	6-2	95,00%	79,00%	83,89%	98,00%
6-1	95,00%	79,33%	84,83%	98,00%	6-2	95,33%	79,00%	83,11%	98,00%
6-1	91,33%	79,67%	83,44%	96,67%	6-2	94,00%	79,00%	83,78%	97,67%
6-1	89,67%	79,00%	83,61%	97,00%	6-2	94,33%	79,33%	84,00%	97,33%
6-1	90,00%	78,33%	82,72%	95,00%	6-2	94,33%	78,33%	83,83%	98,00%
6-1	91,67%	78,00%	83,22%	96,67%	6-2	93,00%	79,00%	83,83%	97,67%
6-1	92,00%	78,33%	83,72%	97,00%	6-2	96,67%	77,67%	84,50%	98,00%
6-1	92,67%	79,67%	83,56%	97,67%	6-2	93,00%	78,67%	82,89%	97,00%
6-1	89,00%	78,33%	83,00%	96,33%	6-2	93,00%	79,67%	83,78%	98,00%
6-1	92,33%	78,33%	83,78%	96,33%	6-2	95,67%	79,33%	84,33%	97,00%
7-1	93,00%	78,67%	84,56%	97,67%	7-2	95,33%	79,67%	84,28%	97,00%
7-1	93,00%	77,67%	84,39%	97,33%	7-2	93,67%	77,67%	83,39%	96,67%
7-1	94,33%	71,67%	81,50%	95,67%	7-2	94,33%	77,67%	84,17%	98,33%
7-1	94,67%	79,67%	84,17%	96,67%	7-2	94,00%	78,67%	84,28%	98,00%
7-1	94,67%	78,67%	83,50%	96,33%	7-2	94,00%	79,67%	84,11%	97,67%
7-1	92,33%	77,33%	83,94%	97,33%	7-2	94,67%	79,33%	84,83%	96,33%
7-1	89,33%	77,33%	82,33%	96,33%	7-2	91,33%	78,33%	83,22%	97,00%
7-1	91,67%	77,67%	83,39%	97,67%	7-2	94,33%	78,67%	83,83%	97,33%
7-1	94,00%	79,00%	84,33%	97,33%	7-2	94,33%	78,33%	84,28%	97,67%
7-1	94,67%	79,33%	84,39%	98,00%	7-2	94,67%	78,33%	84,72%	97,33%

Tabla A.1. Continuación.									
Partición	P <sub>MÁXIMA</sub>	P <sub>MÍNIMA</sub>	P <sub>MEDIA</sub>	P <sub>BCE</sub>	Partición	P <sub>MÁXIMA</sub>	P <sub>MÍNIMA</sub>	P <sub>MEDIA</sub>	P <sub>BCE</sub>
8-1	93,33%	68,00%	80,28%	90,33%	8-2	94,00%	76,00%	83,28%	96,67%
8-1	90,67%	75,67%	81,67%	94,00%	8-2	90,33%	78,67%	83,11%	97,33%
8-1	92,67%	71,67%	81,17%	94,00%	8-2	95,67%	79,33%	83,94%	97,67%
8-1	93,33%	72,67%	81,50%	91,67%	8-2	91,67%	79,00%	84,33%	97,33%
8-1	90,67%	71,33%	81,06%	90,67%	8-2	94,00%	78,67%	83,72%	97,00%
8-1	96,00%	74,00%	82,39%	93,67%	8-2	95,67%	76,67%	83,33%	97,00%
8-1	91,67%	74,67%	82,44%	94,00%	8-2	95,33%	78,33%	83,39%	96,33%
8-1	96,00%	73,33%	82,00%	95,00%	8-2	95,33%	79,00%	84,11%	97,33%
8-1	92,33%	70,33%	80,94%	95,33%	8-2	92,33%	79,00%	83,11%	96,67%
8-1	93,00%	74,33%	82,39%	93,00%	8-2	95,67%	78,67%	83,67%	96,67%
9-1	89,33%	80,33%	84,00%	97,33%	9-2	94,00%	77,33%	84,00%	97,33%
9-1	94,33%	81,00%	85,33%	97,33%	9-2	92,33%	77,67%	83,44%	96,33%
9-1	92,33%	80,33%	84,06%	97,00%	9-2	91,67%	77,00%	83,83%	95,33%
9-1	92,33%	80,00%	84,11%	98,33%	9-2	96,33%	76,00%	84,00%	95,67%
9-1	94,00%	78,67%	84,22%	97,00%	9-2	93,00%	77,00%	84,39%	96,67%
9-1	95,00%	79,00%	85,33%	96,67%	9-2	90,00%	76,67%	82,94%	96,00%
9-1	93,67%	79,67%	85,28%	97,33%	9-2	93,33%	77,00%	83,44%	96,33%
9-1	92,67%	80,33%	84,06%	97,33%	9-2	92,33%	76,67%	83,22%	96,67%
9-1	91,00%	80,67%	83,89%	97,33%	9-2	92,33%	76,33%	83,50%	97,00%
9-1	92,33%	81,00%	85,00%	96,00%	9-2	94,67%	77,00%	83,50%	96,67%
10-1	91,67%	76,67%	82,72%	95,33%	10-2	96,00%	78,67%	83,56%	97,67%
10-1	90,00%	77,00%	82,56%	96,00%	10-2	97,33%	78,00%	83,39%	97,00%
10-1	95,00%	78,00%	83,39%	95,33%	10-2	87,33%	77,67%	81,28%	97,00%
10-1	94,33%	79,00%	84,11%	96,33%	10-2	96,67%	79,00%	83,67%	96,33%
10-1	91,67%	78,67%	83,56%	95,67%	10-2	96,33%	76,67%	82,44%	97,33%
10-1	90,67%	78,00%	82,78%	96,33%	10-2	94,67%	78,00%	82,83%	95,67%
10-1	91,33%	78,67%	82,61%	96,00%	10-2	95,33%	74,33%	82,39%	96,67%
10-1	93,00%	78,00%	82,72%	95,33%	10-2	94,67%	79,33%	83,61%	97,33%
10-1	92,33%	78,00%	82,89%	96,67%	10-2	96,00%	78,33%	83,28%	97,67%
10-1	91,67%	77,67%	82,72%	95,67%	10-2	97,00%	76,33%	82,83%	96,33%
11-1	94,33%	80,00%	85,06%	98,00%	11-2	94,33%	79,00%	84,67%	97,33%
11-1	92,00%	80,33%	84,50%	97,67%	11-2	93,67%	81,33%	84,72%	97,67%
11-1	91,33%	80,00%	83,44%	97,67%	11-2	96,00%	78,67%	84,78%	97,67%
11-1	91,00%	79,67%	83,83%	97,67%	11-2	94,67%	79,33%	84,89%	98,00%
11-1	91,67%	80,00%	84,39%	98,00%	11-2	93,33%	79,33%	83,67%	96,67%
11-1	89,00%	79,67%	83,61%	98,33%	11-2	92,33%	79,00%	83,11%	97,00%
11-1	93,33%	78,33%	84,56%	97,33%	11-2	92,33%	79,00%	84,06%	97,67%
11-1	92,33%	79,33%	84,50%	98,33%	11-2	93,33%	78,33%	84,11%	97,67%
11-1	95,00%	80,00%	84,94%	98,00%	11-2	91,00%	79,67%	84,56%	98,00%
11-1	92,67%	77,00%	84,17%	98,33%	11-2	92,00%	79,00%	83,28%	97,00%
12-1	94,67%	78,33%	83,94%	96,33%	12-2	94,00%	79,67%	83,67%	96,33%
12-1	92,33%	79,67%	84,50%	96,00%	12-2	94,67%	80,00%	84,22%	97,00%
12-1	90,33%	79,33%	83,56%	96,33%	12-2	90,67%	80,00%	82,72%	96,67%
12-1	95,00%	79,33%	84,67%	96,33%	12-2	94,00%	80,33%	84,11%	96,33%
12-1	88,00%	78,33%	81,94%	95,33%	12-2	92,33%	79,67%	83,50%	96,67%
12-1	96,67%	78,67%	84,50%	96,67%	12-2	91,33%	80,00%	83,11%	96,67%
12-1	92,67%	79,00%	83,67%	96,67%	12-2	94,00%	79,00%	83,61%	97,00%
12-1	94,67%	78,00%	84,39%	96,67%	12-2	92,67%	78,33%	82,72%	96,67%
12-1	92,67%	78,00%	82,89%	96,67%	12-2	93,00%	79,67%	83,94%	96,33%
12-1	96,33%	78,33%	84,50%	96,67%	12-2	92,67%	79,00%	83,33%	95,33%

**Tabla A.1. Continuación.**

Partición	P <sub>MÁXIMA</sub>	P <sub>MÍNIMA</sub>	P <sub>MEDIA</sub>	P <sub>BCE</sub>	Partición	P <sub>MÁXIMA</sub>	P <sub>MÍNIMA</sub>	P <sub>MEDIA</sub>	P <sub>BCE</sub>
13-1	91,33%	78,67%	81,78%	93,67%	13-2	94,00%	80,33%	84,67%	96,33%
13-1	93,67%	78,67%	83,28%	94,00%	13-2	91,67%	80,33%	83,61%	96,00%
13-1	92,33%	79,00%	83,44%	95,00%	13-2	95,33%	73,00%	83,72%	95,67%
13-1	90,33%	79,00%	82,33%	94,00%	13-2	95,00%	79,67%	84,67%	96,33%
13-1	<b>95,00%</b>	78,67%	83,33%	94,33%	13-2	94,67%	79,00%	83,89%	95,67%
13-1	93,00%	79,00%	83,56%	95,00%	13-2	95,33%	79,67%	84,83%	95,33%
13-1	92,33%	78,67%	82,28%	95,33%	13-2	97,00%	79,67%	85,39%	98,00%
13-1	92,33%	77,33%	83,00%	94,33%	13-2	93,00%	80,33%	84,61%	95,67%
13-1	93,00%	79,00%	83,11%	94,67%	13-2	94,67%	80,00%	84,61%	97,67%
13-1	92,00%	79,67%	83,33%	95,00%	13-2	95,00%	80,00%	84,67%	96,00%
14-1	91,00%	79,00%	82,61%	96,67%	14-2	94,00%	80,00%	84,56%	96,67%
14-1	95,00%	77,67%	84,44%	97,33%	14-2	94,67%	80,33%	84,72%	96,67%
14-1	96,00%	79,33%	84,50%	97,00%	14-2	92,67%	80,00%	84,50%	97,33%
14-1	94,33%	79,00%	84,17%	96,33%	14-2	95,00%	79,33%	84,72%	96,67%
14-1	94,67%	79,33%	84,94%	97,33%	14-2	95,67%	80,67%	84,94%	97,33%
14-1	94,00%	78,33%	83,67%	96,67%	14-2	95,00%	80,00%	84,67%	97,67%
14-1	93,67%	77,33%	83,39%	96,67%	14-2	94,67%	80,33%	85,33%	97,00%
14-1	94,67%	77,67%	83,00%	96,00%	14-2	92,33%	79,33%	83,61%	97,00%
14-1	95,67%	79,67%	85,50%	98,33%	14-2	95,33%	79,00%	84,78%	97,67%
14-1	96,33%	77,67%	84,06%	97,00%	14-2	93,67%	80,00%	83,83%	97,33%
15-1	93,00%	76,00%	85,00%	98,00%	15-2	89,67%	77,33%	82,50%	95,33%
15-1	93,67%	77,33%	84,72%	98,00%	15-2	92,00%	78,33%	83,72%	96,33%
15-1	94,00%	77,67%	83,83%	98,00%	15-2	94,67%	77,33%	83,78%	96,00%
15-1	94,33%	77,33%	84,61%	97,33%	15-2	92,00%	77,33%	83,33%	96,00%
15-1	94,67%	78,00%	85,33%	97,00%	15-2	91,67%	75,33%	83,39%	96,00%
15-1	95,00%	76,67%	84,89%	97,67%	15-2	90,67%	77,00%	83,33%	96,00%
15-1	95,33%	77,67%	85,39%	97,00%	15-2	91,00%	76,67%	83,06%	95,33%
15-1	94,33%	77,33%	84,72%	97,33%	15-2	92,33%	77,33%	83,89%	95,67%
15-1	94,00%	77,67%	85,56%	97,67%	15-2	92,00%	75,67%	82,78%	95,67%
15-1	94,33%	76,33%	85,06%	98,00%	15-2	93,33%	76,67%	83,83%	96,00%
16-1	91,67%	80,00%	83,44%	95,33%	16-2	97,67%	80,00%	85,00%	98,00%
16-1	95,33%	79,67%	84,56%	96,67%	16-2	96,33%	79,67%	83,22%	97,33%
16-1	94,33%	78,33%	84,06%	97,33%	16-2	95,67%	80,67%	84,28%	97,67%
16-1	93,33%	79,67%	84,22%	97,33%	16-2	<b>97,67%</b>	80,00%	83,94%	97,33%
16-1	96,33%	79,33%	84,28%	97,00%	16-2	94,67%	79,00%	82,89%	96,67%
16-1	92,00%	79,67%	83,72%	96,67%	16-2	96,33%	79,33%	84,00%	98,33%
16-1	93,67%	79,00%	83,44%	96,67%	16-2	97,67%	79,33%	83,61%	98,33%
16-1	<b>96,67%</b>	79,00%	84,33%	96,33%	16-2	97,67%	80,00%	83,94%	98,00%
16-1	<b>97,00%</b>	79,00%	84,22%	96,00%	16-2	96,33%	80,67%	84,67%	98,33%
16-1	94,67%	78,67%	83,78%	96,00%	16-2	97,33%	79,67%	84,00%	98,67%
17-1	90,00%	78,00%	83,56%	96,00%	17-2	96,33%	79,00%	84,83%	97,67%
17-1	93,33%	77,33%	83,11%	96,67%	17-2	95,33%	79,33%	84,22%	97,33%
17-1	92,33%	78,33%	84,00%	96,33%	17-2	95,67%	78,33%	83,33%	98,00%
17-1	94,00%	78,33%	84,83%	96,67%	17-2	95,33%	80,00%	83,39%	97,67%
17-1	92,33%	78,33%	84,28%	96,67%	17-2	97,00%	80,00%	84,56%	97,67%
17-1	90,67%	78,33%	83,22%	96,67%	17-2	95,67%	79,33%	84,39%	96,00%
17-1	91,67%	77,67%	84,11%	96,67%	17-2	96,67%	78,33%	83,33%	97,67%
17-1	95,00%	78,33%	84,89%	96,33%	17-2	94,33%	78,33%	83,33%	97,00%
17-1	89,67%	79,00%	83,22%	97,00%	17-2	97,00%	79,67%	84,17%	97,33%
17-1	93,67%	78,00%	84,22%	97,00%	17-2	97,00%	79,00%	84,06%	98,00%

Tabla A.1. Continuación.									
Partición	P <sub>MÁXIMA</sub>	P <sub>MÍNIMA</sub>	P <sub>MEDIA</sub>	P <sub>BCE</sub>	Partición	P <sub>MÁXIMA</sub>	P <sub>MÍNIMA</sub>	P <sub>MEDIA</sub>	P <sub>BCE</sub>
18-1	96,33%	80,33%	84,89%	98,00%	18-2	92,00%	77,67%	83,22%	95,33%
18-1	93,00%	77,00%	82,22%	94,67%	18-2	91,33%	79,33%	83,33%	97,33%
18-1	95,33%	79,33%	83,00%	96,33%	18-2	96,33%	78,00%	84,61%	97,33%
18-1	95,33%	79,67%	83,72%	97,67%	18-2	90,00%	78,67%	82,39%	95,67%
18-1	96,67%	80,67%	84,61%	96,67%	18-2	92,33%	77,67%	83,44%	96,00%
18-1	90,67%	78,67%	83,11%	94,33%	18-2	95,33%	77,00%	83,28%	96,00%
18-1	93,00%	79,33%	83,56%	95,33%	18-2	95,00%	79,33%	83,83%	96,33%
18-1	92,67%	80,00%	82,78%	95,00%	18-2	93,00%	78,00%	83,61%	96,00%
18-1	89,00%	80,00%	82,56%	96,00%	18-2	94,67%	77,33%	84,06%	96,67%
18-1	91,33%	80,00%	82,67%	94,33%	18-2	92,67%	78,33%	83,56%	96,00%
19-1	93,33%	77,33%	83,39%	94,33%	19-2	94,67%	79,67%	84,83%	98,00%
19-1	91,33%	78,33%	83,56%	95,33%	19-2	95,00%	80,33%	85,28%	98,00%
19-1	93,67%	77,33%	83,61%	95,67%	19-2	93,00%	78,67%	85,17%	98,00%
19-1	96,00%	78,33%	84,28%	94,67%	19-2	94,33%	81,00%	85,89%	98,33%
19-1	91,33%	78,33%	83,33%	95,67%	19-2	95,33%	80,67%	85,44%	97,67%
19-1	91,00%	78,00%	83,11%	95,67%	19-2	94,67%	81,00%	84,67%	97,67%
19-1	95,00%	77,00%	83,94%	95,00%	19-2	95,00%	80,33%	85,00%	98,33%
19-1	91,67%	78,33%	83,17%	94,67%	19-2	95,33%	80,67%	84,50%	98,00%
19-1	96,00%	78,00%	83,94%	95,00%	19-2	95,00%	80,67%	84,61%	98,00%
19-1	92,67%	77,33%	83,28%	94,67%	19-2	89,67%	80,67%	84,33%	98,00%
20-1	90,67%	78,33%	83,72%	96,33%	20-2	92,67%	79,00%	83,22%	97,33%
20-1	90,33%	78,33%	82,83%	96,33%	20-2	92,67%	80,67%	84,44%	98,00%
20-1	89,00%	78,33%	82,61%	96,00%	20-2	91,67%	80,67%	83,56%	97,00%
20-1	92,67%	78,33%	83,44%	96,33%	20-2	95,33%	80,33%	84,94%	97,00%
20-1	92,33%	79,67%	84,06%	97,00%	20-2	93,67%	80,33%	83,72%	98,33%
20-1	92,33%	78,00%	83,06%	97,00%	20-2	95,33%	80,00%	84,83%	97,33%
20-1	91,00%	77,67%	83,11%	97,00%	20-2	93,00%	80,33%	83,78%	97,00%
20-1	90,67%	76,33%	83,06%	95,33%	20-2	91,33%	80,33%	83,00%	98,33%
20-1	90,67%	77,00%	83,11%	96,67%	20-2	92,00%	80,33%	84,06%	97,67%
20-1	91,00%	77,67%	83,50%	96,67%	20-2	93,33%	79,33%	84,00%	97,33%
21-1	94,00%	79,67%	84,72%	96,00%	21-2	96,00%	78,33%	84,78%	97,33%
21-1	97,33%	80,00%	85,56%	96,00%	21-2	95,33%	78,33%	83,39%	96,33%
21-1	96,67%	78,67%	85,94%	96,33%	21-2	97,00%	79,00%	85,83%	96,33%
21-1	96,33%	79,67%	85,28%	97,00%	21-2	94,67%	78,67%	84,56%	97,33%
21-1	95,67%	79,33%	85,17%	95,67%	21-2	96,33%	79,00%	85,28%	97,33%
21-1	96,00%	79,33%	85,00%	95,00%	21-2	96,00%	78,33%	85,61%	97,00%
21-1	95,00%	79,33%	84,89%	95,00%	21-2	94,00%	78,67%	83,67%	96,33%
21-1	96,33%	80,00%	85,78%	96,33%	21-2	96,67%	79,00%	85,56%	97,67%
21-1	97,00%	80,00%	84,83%	96,33%	21-2	95,33%	78,67%	84,67%	96,33%
21-1	95,67%	79,33%	85,89%	96,33%	21-2	93,67%	78,00%	84,72%	97,33%
22-1	95,67%	78,00%	85,78%	96,67%	22-2	93,67%	76,00%	83,50%	96,33%
22-1	96,67%	77,67%	84,78%	97,00%	22-2	94,33%	76,33%	84,28%	95,00%
22-1	96,00%	78,00%	85,61%	97,33%	22-2	94,00%	74,33%	83,67%	94,00%
22-1	94,33%	77,33%	85,56%	97,00%	22-2	93,67%	72,00%	83,44%	96,00%
22-1	95,33%	78,33%	85,78%	97,67%	22-2	94,33%	72,33%	83,67%	94,33%
22-1	96,67%	77,67%	86,06%	98,33%	22-2	96,00%	75,67%	85,28%	96,00%
22-1	94,33%	78,00%	85,33%	96,67%	22-2	94,67%	76,00%	83,11%	95,00%
22-1	95,67%	79,67%	85,33%	97,67%	22-2	95,33%	74,33%	83,72%	95,67%
22-1	95,00%	78,00%	86,22%	97,67%	22-2	93,33%	72,33%	83,11%	94,67%
22-1	96,00%	78,00%	86,56%	97,67%	22-2	93,67%	75,33%	83,50%	95,67%



**Tabla A.1. Continuación.**

Partición	P <sub>MÁXIMA</sub>	P <sub>MÍNIMA</sub>	P <sub>MEDIA</sub>	P <sub>BCE</sub>	Partición	P <sub>MÁXIMA</sub>	P <sub>MÍNIMA</sub>	P <sub>MEDIA</sub>	P <sub>BCE</sub>
23-1	96,33%	79,00%	85,17%	97,33%	23-2	96,67%	81,00%	86,22%	97,33%
23-1	96,00%	77,33%	84,00%	96,33%	23-2	97,00%	81,33%	86,28%	97,33%
23-1	95,33%	79,33%	84,83%	96,67%	23-2	96,67%	81,00%	86,22%	97,33%
23-1	96,33%	78,67%	84,56%	96,67%	23-2	96,00%	80,00%	86,00%	97,00%
23-1	95,67%	79,00%	85,22%	97,67%	23-2	95,33%	80,67%	85,50%	98,00%
23-1	96,67%	76,33%	84,56%	96,67%	23-2	97,00%	81,67%	86,44%	97,67%
23-1	96,33%	78,33%	84,50%	96,33%	23-2	96,67%	82,00%	86,78%	97,67%
23-1	<b>96,67%</b>	78,00%	84,28%	96,00%	23-2	97,33%	82,00%	87,50%	98,00%
23-1	95,67%	78,00%	84,17%	97,33%	23-2	96,67%	82,00%	86,61%	98,00%
23-1	96,67%	78,00%	85,61%	97,00%	23-2	96,67%	82,00%	86,22%	97,33%
24-1	<b>95,33%</b>	77,00%	84,00%	93,67%	24-2	95,67%	76,67%	84,83%	96,67%
24-1	<b>94,67%</b>	80,00%	84,78%	94,33%	24-2	<b>97,33%</b>	78,00%	85,83%	96,67%
24-1	94,00%	79,00%	85,28%	95,00%	24-2	95,33%	75,67%	85,22%	97,33%
24-1	<b>94,33%</b>	78,67%	84,78%	94,00%	24-2	96,67%	76,00%	84,83%	96,67%
24-1	94,67%	78,67%	84,00%	95,00%	24-2	<b>96,33%</b>	78,33%	84,56%	96,00%
24-1	94,00%	79,00%	84,83%	94,33%	24-2	96,33%	77,00%	84,61%	97,67%
24-1	<b>95,33%</b>	78,00%	84,67%	95,00%	24-2	<b>96,00%</b>	75,00%	85,28%	95,67%
24-1	<b>95,67%</b>	78,33%	84,61%	94,67%	24-2	96,33%	77,00%	85,28%	97,33%
24-1	96,00%	78,33%	85,61%	96,00%	24-2	96,67%	77,33%	84,72%	96,67%
24-1	95,00%	78,33%	84,33%	95,33%	24-2	<b>96,67%</b>	76,67%	85,28%	95,67%
25-1	95,33%	77,33%	84,78%	95,33%	25-2	95,00%	79,67%	85,00%	97,33%
25-1	94,67%	79,33%	85,39%	96,67%	25-2	96,33%	80,33%	85,22%	98,33%
25-1	94,33%	78,67%	84,44%	96,33%	25-2	97,00%	80,00%	86,67%	98,00%
25-1	95,00%	72,00%	83,94%	96,00%	25-2	94,67%	79,00%	86,00%	98,00%
25-1	95,00%	79,67%	85,56%	97,00%	25-2	96,33%	80,33%	85,72%	98,00%
25-1	94,00%	79,33%	84,50%	95,33%	25-2	96,67%	79,00%	85,78%	98,00%
25-1	93,33%	78,33%	83,78%	95,33%	25-2	94,33%	79,33%	84,61%	98,00%
25-1	95,33%	79,00%	84,22%	96,33%	25-2	97,67%	80,33%	87,17%	98,33%
25-1	93,67%	79,33%	84,94%	96,33%	25-2	95,33%	79,67%	85,33%	98,33%
25-1	96,33%	78,33%	85,17%	97,00%	25-2	96,00%	80,00%	85,94%	97,67%
26-1	<b>97,33%</b>	75,00%	84,11%	95,00%	26-2	94,67%	78,33%	85,00%	95,67%
26-1	<b>96,00%</b>	75,00%	82,78%	94,33%	26-2	95,33%	77,33%	84,78%	96,00%
26-1	95,00%	76,00%	84,06%	95,00%	26-2	<b>96,67%</b>	78,33%	85,17%	96,33%
26-1	95,00%	75,67%	83,78%	95,33%	26-2	95,67%	77,33%	84,56%	95,67%
26-1	<b>95,67%</b>	75,00%	84,44%	95,33%	26-2	96,00%	69,67%	83,78%	96,67%
26-1	94,33%	75,33%	83,50%	94,33%	26-2	95,00%	77,00%	84,61%	96,00%
26-1	95,33%	75,00%	84,61%	95,67%	26-2	<b>96,00%</b>	76,67%	84,11%	95,00%
26-1	94,67%	76,00%	83,39%	94,67%	26-2	96,00%	77,67%	85,28%	96,33%
26-1	<b>95,67%</b>	75,00%	83,72%	93,33%	26-2	94,33%	77,67%	84,33%	95,67%
26-1	<b>94,67%</b>	75,00%	83,39%	93,67%	26-2	94,33%	78,00%	84,44%	95,00%
27-1	94,67%	79,33%	85,67%	95,67%	27-2	95,33%	78,00%	84,33%	97,67%
27-1	93,00%	78,33%	84,56%	95,00%	27-2	97,00%	80,67%	85,61%	98,33%
27-1	93,67%	77,67%	84,17%	95,33%	27-2	96,67%	81,00%	86,06%	98,00%
27-1	94,00%	78,67%	84,17%	95,33%	27-2	96,33%	80,33%	85,78%	99,00%
27-1	94,67%	78,67%	84,61%	95,67%	27-2	96,67%	81,00%	86,56%	98,67%
27-1	93,00%	79,00%	83,94%	95,33%	27-2	96,33%	80,00%	86,44%	97,33%
27-1	94,33%	77,33%	84,33%	94,67%	27-2	97,00%	82,00%	86,17%	98,67%
27-1	94,33%	79,33%	84,83%	95,00%	27-2	95,33%	80,67%	86,22%	97,00%
27-1	93,00%	79,33%	83,83%	95,00%	27-2	95,33%	74,67%	84,78%	97,33%
27-1	94,33%	78,00%	84,44%	94,33%	27-2	97,00%	81,67%	86,39%	98,33%



Tabla A.1. Continuación.									
Partición	P <sub>MÁXIMA</sub>	P <sub>MÍNIMA</sub>	P <sub>MEDIA</sub>	P <sub>BCE</sub>	Partición	P <sub>MÁXIMA</sub>	P <sub>MÍNIMA</sub>	P <sub>MEDIA</sub>	P <sub>BCE</sub>
28-1	94,67%	79,00%	84,39%	96,00%	28-2	94,67%	78,33%	84,56%	96,00%
28-1	<b>97,33%</b>	78,33%	85,78%	96,00%	28-2	94,33%	80,33%	84,94%	96,00%
28-1	96,00%	79,33%	85,39%	97,00%	28-2	96,33%	80,00%	85,72%	97,00%
28-1	95,33%	79,33%	84,83%	95,67%	28-2	95,67%	80,67%	84,56%	96,67%
28-1	<b>97,67%</b>	78,33%	84,94%	96,33%	28-2	93,67%	80,00%	85,22%	96,33%
28-1	95,33%	79,33%	85,56%	96,67%	28-2	96,00%	80,00%	85,94%	96,67%
28-1	95,33%	78,67%	85,28%	96,67%	28-2	95,33%	80,33%	84,89%	97,00%
28-1	<b>96,67%</b>	79,33%	85,67%	95,67%	28-2	94,67%	80,33%	84,56%	95,67%
28-1	96,00%	79,33%	84,61%	96,67%	28-2	96,00%	80,00%	85,61%	96,00%
28-1	95,33%	78,00%	85,28%	95,67%	28-2	94,00%	80,00%	85,06%	96,67%
29-1	93,00%	77,33%	84,22%	95,33%	29-2	93,67%	78,33%	84,33%	96,67%
29-1	92,33%	77,00%	84,11%	95,33%	29-2	96,00%	76,00%	85,44%	96,33%
29-1	93,67%	77,33%	84,17%	94,67%	29-2	94,33%	77,00%	84,72%	96,00%
29-1	<b>96,00%</b>	79,67%	84,72%	95,67%	29-2	94,33%	78,00%	85,17%	96,33%
29-1	<b>96,67%</b>	78,00%	85,00%	95,67%	29-2	95,67%	76,67%	85,33%	97,00%
29-1	<b>96,33%</b>	78,00%	85,17%	95,33%	29-2	<b>96,67%</b>	77,33%	85,22%	96,33%
29-1	<b>97,00%</b>	77,67%	85,00%	94,33%	29-2	95,67%	78,00%	85,22%	96,67%
29-1	<b>96,67%</b>	77,33%	84,11%	95,33%	29-2	95,00%	77,67%	86,06%	96,00%
29-1	95,00%	77,33%	84,39%	96,00%	29-2	<b>97,33%</b>	78,33%	85,50%	96,33%
29-1	92,33%	77,67%	83,22%	94,33%	29-2	95,33%	77,33%	85,72%	96,00%
30-1	95,00%	78,00%	84,94%	97,00%	30-2	96,33%	78,67%	85,44%	97,33%
30-1	95,67%	78,33%	85,50%	97,00%	30-2	95,67%	79,00%	84,50%	96,33%
30-1	97,00%	78,67%	86,28%	97,00%	30-2	95,00%	79,33%	84,39%	96,00%
30-1	96,67%	78,33%	86,17%	97,00%	30-2	92,67%	78,33%	84,39%	95,67%
30-1	<b>96,00%</b>	79,33%	85,83%	95,33%	30-2	93,33%	78,67%	83,78%	96,00%
30-1	<b>98,33%</b>	78,67%	85,83%	97,00%	30-2	94,33%	80,00%	84,67%	96,67%
30-1	94,00%	78,67%	85,22%	97,33%	30-2	95,67%	79,33%	85,11%	96,67%
30-1	<b>96,33%</b>	78,67%	85,67%	95,67%	30-2	93,00%	77,33%	84,22%	96,00%
30-1	96,33%	80,00%	85,94%	97,00%	30-2	92,67%	77,00%	83,28%	96,33%
30-1	95,67%	79,67%	85,06%	97,33%	30-2	92,33%	77,33%	83,50%	95,67%



## Apéndice B

### Estadísticos para la Clasificación de Textos

En este apéndice se detalla la base teórica y el cálculo de los estadísticos usados para comparar los clasificadores usados en la Clasificación de Textos.

#### Micro sign test (*s-test*):

Este test compara dos clasificadores, *A* y *B*, analizando la clase asignada por cada uno de los sistemas a los distintos documentos que componen el conjunto de test.

Si:

*M* es el número de decisiones binarias de cada sistemas (producto del número de documentos de test y el número de categorías),

$a_i \in \{0,1\}$  indica si el sistema *A* realiza una decisión correcta (1) o incorrecta (0)  $i=1, 2.. N$

$b_i \in \{0,1\}$  indica si el sistema *B* realiza una decisión correcta (1) o incorrecta (0)

*n* es el número de veces que  $a_i$  y  $b_i$  difieren

*k* es el número de veces que  $a_i$  es mayor que  $b_i$  ( $a_i=1$ ;  $b_i=0$ )

entonces, ambos sistemas serán equivalentes si, en aquellos casos en los que sus decisiones difieren, la probabilidad de acierto de cada uno de ellos es 0.5. Es decir, ambos sistemas serán equivalentes si cada uno acierta por separado  $n/2$  veces. En [Yang & Liu, 1999] se asume que para valores de  $n>12$ , el número de ejemplos correctamente clasificados por *A* sigue una distribución normal (Figura B.1) con media  $\mu=n/2$  y desviación estándar  $\sigma=0.5$ . Bajo estos supuesto, se podrá afirmar que el sistema *A* es mejor que el sistema *B* ( $\mu>0.5$ ) con un nivel de confianza  $(1-\alpha)\%$  si:

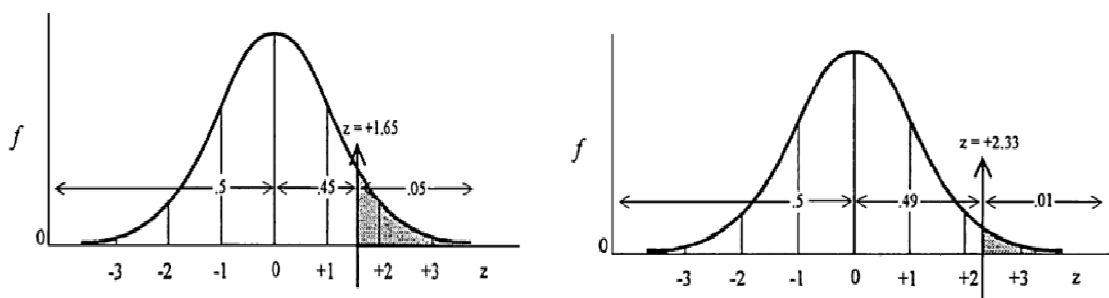
$$Z = \frac{k-0.5n}{0.5\sqrt{n}} \quad [\text{Ec. B.1}]$$

es positivo y su valor es mayor o igual que el *valor normal tipificado* que deja a su derecha un área  $\alpha$  ( $Z_\alpha$ ).

En la Tabla B.1. se recoge el valor normal tipificado o valor crítico para los valores de  $\alpha$  más habituales.

**Tabla B.1. Valor normal tipificado para  $\alpha=0,05$  y  $\alpha=0,01$ .**

$Z_{0,05}$	$Z_{0,01}$
1,65	2,33



**Figura B.1. Curva Normal Tipificada donde se señalan los valores de  $Z$  que dejan a su derecha un área igual a 0,05 (izquierda) y 0,01 (derecha).**

### Macro sign test (*S-test*):

Este test compara dos clasificadores,  $A$  y  $B$ , analizando el valor de la medida  $F_I$  [Ec. B.2] de cada uno de los sistemas sobre cada una de las categorías asociadas al conjunto de test:

$$F_1^i = \frac{2TP_i}{2TP_i + FP_i + FN_i} \quad [\text{Ec. B.2}]$$

Si:

$M$  es el número de categorías (clases)

$a_i \in [0,1]$  es el valor de  $F_I$  obtenido por el sistema  $A$  sobre la categoría  $i$  ( $i=1, 2.. M$ )

$b_i \in [0,1]$  es el valor de  $F_I$  obtenido por el sistema  $B$  sobre la categoría  $i$  ( $i=1, 2.. M$ )

$n$  es el número de veces que  $a_i$  y  $b_i$  difieren. (A efectos prácticos se considera que los valores de  $F_I$  coinciden si la diferencia entre ambos es menor de 0,001)

$k$  es el número de veces que  $a_i$  es mayor que  $b_i$

los sistemas  $A$  y  $B$  se considerarán equivalentes cuando  $k=0,5n$ .

Al igual que en el caso anterior, para valores de  $n > 12$ , el número de casos en los que  $a_i$  es mayor que  $b_i$ , sigue una *distribución normal* con media  $\mu = n/2$  y desviación estándar  $\sigma = 0,5$ . Por tanto, para determinar si el sistema A es mejor que el sistema B bastará con calcular el valor Z aplicando la Ecuación B.1, y compararlo con los valores recogidos en la Tabla B.1.

### Macro t-test (T-test):

Dado que los valores de  $F_1$  obtenidos por dos sistemas se calculan sobre el mismo conjunto de datos, estos valores se pueden considerar como dos muestras *dependientes* que pueden o no representar poblaciones con diferentes valores medios. Así, admitiendo la notación definida para  $n^{17}$ ,  $a_i$  y  $b_i$  en el punto anterior, considerando que:

$d_i = a_i - b_i$  es la diferencia entre  $a_i$  y  $b_i$ ,

$\bar{d}$  es la media de los valores de  $d_i$  para  $i = 1, 2, \dots, n$ ,

$$s_{\bar{d}} = \frac{s}{\sqrt{n}} = \frac{\sqrt{\frac{1}{n-1} \sum_{i=1}^n (d_i - \bar{d})^2}}{\sqrt{n}} \text{ representa el error estándar de la media,}$$

y asumiendo que los valores de  $F_1$  dados por cada uno de los sistemas están *normalmente distribuidos*, se puede concluir que la variable aleatoria,

$$t = \frac{\bar{d}}{s_{\bar{d}}} \quad [\text{Ec. B.3}]$$

sigue una distribución *t de Student* con  $n-1$  grados de libertad [Sheskin, 2000].

Por tanto, se podrá afirmar que el sistema A es mejor que el sistema B con un nivel de confianza del  $(1-\alpha)\%$  siempre que el valor de  $t$  sea positivo y mayor o igual que el valor para el que la distribución *t de Student* con  $(n-1)$  grados de libertad deja a su derecha un área  $\alpha$  (valor *t crítico*).

Dado que, cuando el número de grados de libertad de la *distribución t* es elevado, la curva de esta distribución se aproxima estrechamente a la curva de la *distribución normal*, en [Yang & Liu, 1999] se indica que para  $n > 40$  el valor *t crítico* se puede calcular usando la *distribución normal*. Sin embargo en [Álvarez, 2007] se señala que esta aproximación es un resto de épocas en las que no había ayudas electrónicas al cálculo y recomiendan prescindir de ella. Por este motivo, y en lo que a la aplicación del **Macro t-test** se refiere, en esta Tesis los cálculos se realizarán usando la *distribución t de Student* y no la *distribución normal tipificada*. En nuestras aplicaciones, el valor crítico se obtiene aplicando en una *hoja de cálculo* la función DISTR.T.INV ( $2\alpha, n-1$ ) donde el valor de  $\alpha$  se duplica con la finalidad de asociar los resultados a distribuciones

<sup>17</sup> En [Yang & Liu, 1999] los cálculos se realizan considerando  $n$  como el número de veces en los que  $a_i$  y  $b_i$  difieren. Sin embargo en [Sheskin, 2000] (pag. 454) los cálculos se realizan considerando el tamaño de la población que en este caso, y según la nomenclatura utilizada, sería  $M$ . En este trabajo de Tesis se seguirán las pautas indicadas en [Yang & Liu, 1999].

de una cola pues, por defecto, los valores devueltos por la función DISTR.T.INV están referidos a distribuciones de dos colas.

### Diferencias de proporciones: p-test.

Sean  $P_1$  y  $P_2$  las proporciones de éxito observadas en dos grandes muestras de tamaños  $n_1$  y  $n_2$ . Si se admite que ambas muestras han sido extraídas de una misma población, la *distribución muestral* de la diferencia de proporciones se distribuirá aproximadamente como una *normal* de media 0 y *desviación típica*:

$$s = \sqrt{p(1-p)\left(\frac{1}{n_1} + \frac{1}{n_2}\right)} \quad [\text{Ec. B.4}]$$

donde:

$$p = \frac{n_1 P_1 + n_2 P_2}{n_1 + n_2} \quad [\text{Ec. B.5}]$$

es el valor estimado de la proporción poblacional [Spiegel, 1976].

Bajo estas condiciones, la variable tipificada:

$$Z = \frac{P_1 - P_2}{\sqrt{p(1-p)\left(\frac{1}{n_1} + \frac{1}{n_2}\right)}} \quad [\text{Ec. B.6}]$$

permite ensayar las diferencias observadas a un nivel de significación apropiado y, de ese modo, contrastar, entre otras, la hipótesis  $p_1=p_2$  que indica que ambas muestras provienen de la misma población y, por tanto, que no existen diferencias significativas entre ambas.

Dado que:

- la *precisión* de un clasificador se puede ver como la proporción de éxito observada sobre el total de ejemplos a los que se le ha asignado una categoría ( $TP+FP$ ) y
- la *sensibilidad* -en inglés *recall*- como la proporción de éxito observada sobre el total de clasificaciones que se deben realizar ( $TP+FN$ ),

este test permite comparar dos sistemas  $A$  y  $B$  analizando los valores de *precisión* o *sensibilidad* obtenidos para cada uno de ellos.

Al igual que en los casos anteriores se dirá que el clasificador  $A$  es mejor que el clasificador  $B$  con un nivel de confianza del  $(1-\alpha)\%$  si el valor de  $Z$  [Ec. B.6] es positivo y mayor o igual que el valor normal tipificado recogido en la Tabla B.1.

## **Apéndice C**

### **LYRL2004: Resultados Experimentales**

En este apéndice se muestran las tablas de contingencia obtenidas al evaluar los modelos descritos en el epígrafe 5.3.2 sobre la base de datos LYRL2004.

**Tabla C.1. Valores obtenidos al evaluar el Modelo1 (ANN) sobre la base de datos LYRL2004.**

CLASE	TP	FP	FN	TN	$\pi$	$\rho$	$F_1$
CCAT	340614	31391	29927	379333	0,916	0,919	0,917
C11	7439	7420	16212	750194	0,501	0,315	0,386
C12	6363	2886	5200	766816	0,688	0,550	0,611
C13	15509	17641	20954	727161	0,468	0,425	0,446
C14	3833	3057	3417	770958	0,556	0,529	0,542
C15	126837	11542	20769	622117	0,917	0,859	0,887
C151	67788	4689	11736	697052	0,935	0,852	0,892
C1511	17654	2776	5158	755677	0,864	0,774	0,817
C152	52895	12130	18267	697973	0,813	0,743	0,777
C16	588	218	1283	779176	0,730	0,314	0,439
C17	26778	10274	14205	730008	0,723	0,653	0,686
C171	10845	5573	7031	757816	0,661	0,607	0,632
C172	7808	2918	3394	767145	0,728	0,697	0,712
C173	1164	1017	1396	777688	0,534	0,455	0,491
C174	4508	481	1117	775159	0,904	0,801	0,849
C18	41086	15445	10269	714465	0,727	0,800	0,762
C181	32973	14451	9196	724645	0,695	0,782	0,736
C182	1446	1878	3083	774858	0,435	0,319	0,368
C183	3624	1804	3580	772257	0,668	0,503	0,574
C21	9968	7260	14642	749395	0,579	0,405	0,477
C22	2000	1924	3929	773412	0,510	0,337	0,406
C23	812	598	1751	778104	0,576	0,317	0,409
C24	16157	12486	15074	737548	0,564	0,517	0,540
C31	18146	12245	21305	729569	0,597	0,460	0,520
C311	2080	4855	2053	772277	0,300	0,503	0,376
C312	2938	8390	3514	766423	0,259	0,455	0,330
C313	172	2248	902	777943	0,071	0,160	0,098
C32	418	118	1623	779106	0,780	0,205	0,324
C33	7285	5713	7604	760663	0,560	0,489	0,522
C331	241	41	938	780045	0,855	0,204	0,330
C34	1827	1436	2888	775114	0,560	0,387	0,458
C41	8177	2053	2866	768169	0,799	0,740	0,769
C411	7972	2226	2014	769053	0,782	0,798	0,790
C42	7698	2842	3837	766888	0,730	0,667	0,697
ECAT	87443	17263	29028	647531	0,835	0,751	0,791
E11	4102	3292	4187	769684	0,555	0,495	0,523
E12	12902	8378	13519	746466	0,606	0,488	0,541
E121	1281	493	807	778684	0,722	0,614	0,663
E13	4796	1755	1620	773094	0,732	0,748	0,740
E131	4065	1992	1427	773781	0,671	0,740	0,704
E132	358	64	564	780279	0,848	0,388	0,533
E14	846	647	1266	778506	0,567	0,401	0,469
E141	109	92	255	780809	0,542	0,299	0,386
E142	10	5	182	781068	0,667	0,052	0,097
E143	573	333	599	779760	0,632	0,489	0,551
E21	29721	6390	12154	733000	0,823	0,710	0,762
E211	8881	3968	6480	761936	0,691	0,578	0,630
E212	19865	3740	6687	750973	0,842	0,748	0,792
E31	1030	235	1319	778681	0,814	0,438	0,570
E311	827	394	831	779213	0,677	0,499	0,575
E313	3	5	105	781152	0,375	0,028	0,052



Tabla C.1. Continuación							
CLASE	TP	FP	FN	TN	$\pi$	$\rho$	$F_1$
E41	10510	2372	6076	762307	0,816	0,634	0,713
E411	122	54	1974	779115	0,693	0,058	0,107
E51	9887	7530	10752	753096	0,568	0,479	0,520
E511	1145	1880	1686	776554	0,379	0,404	0,391
E512	6090	5643	6144	763388	0,519	0,498	0,508
E513	1160	303	1076	778726	0,793	0,519	0,627
E61	39	24	337	780865	0,619	0,104	0,178
E71	4593	338	509	775825	0,931	0,900	0,916
GCAT	210906	21510	21391	527458	0,907	0,908	0,908
G15	14862	4954	5447	756002	0,750	0,732	0,741
G151	533	869	2725	777138	0,380	0,164	0,229
G152	263	1104	1809	778089	0,192	0,127	0,153
G153	867	535	1434	778429	0,618	0,377	0,468
G154	6178	2191	2088	770808	0,738	0,747	0,743
G155	279	448	1807	778731	0,384	0,134	0,198
G156	0	0	258	781007	0,000	0,000	0,000
G157	377	116	1614	779158	0,765	0,189	0,304
G158	1009	424	3239	776593	0,704	0,238	0,355
G159	0	0	38	781227	0,000	0,000	0,000
GCRIM	21411	9982	9675	740197	0,682	0,689	0,685
GDEF	3879	3238	4730	769418	0,545	0,451	0,493
GDIP	19336	8266	17399	736264	0,701	0,526	0,601
GDIS	4770	1892	3594	771009	0,716	0,570	0,635
GENT	1777	1625	1918	775945	0,522	0,481	0,501
GENV	1928	1932	4161	773244	0,499	0,317	0,388
GFAS	0	0	307	780958	0,000	0,000	0,000
GHEA	2875	2778	2958	772654	0,509	0,493	0,501
GJOB	11867	5496	4903	758999	0,683	0,708	0,695
GOBIT	0	0	831	780434	0,000	0,000	0,000
GODD	229	948	2483	777605	0,195	0,084	0,118
GPOL	36800	23279	18431	702755	0,613	0,666	0,638
GPRO	1169	1365	4163	774568	0,461	0,219	0,297
GREL	1085	579	1672	777929	0,652	0,394	0,491
GSCI	1150	424	1223	778468	0,731	0,485	0,583
GSPO	32676	1401	1728	745460	0,959	0,950	0,954
GTOUR	178	67	479	780541	0,727	0,271	0,395
GVIO	20032	9262	11468	740503	0,684	0,636	0,659
GVOTE	6796	5417	4390	764662	0,556	0,608	0,581
GWEA	1865	654	1878	776868	0,740	0,498	0,596
GWELF	271	128	1547	779319	0,679	0,149	0,244
MCAT	183847	25267	15091	557060	0,879	0,924	0,901
M11	42715	8526	4687	725337	0,834	0,901	0,866
M12	20600	7889	4704	748072	0,723	0,814	0,766
M13	43967	9205	8071	720022	0,827	0,845	0,836
M131	22218	4844	5024	749179	0,821	0,816	0,818
M132	20631	5927	5422	749285	0,777	0,792	0,784
M14	74880	8766	8019	689600	0,895	0,903	0,899
M141	42555	7049	3645	728016	0,858	0,921	0,888
M142	9108	1263	2711	768183	0,878	0,771	0,821
M143	18893	3547	2458	756367	0,842	0,885	0,863
Micro-media	1950783	472383	582338	75902261	0,805	0,770	0,787
Macro-media					0,639	0,515	0,550

**Tabla C.2. Valores obtenidos al evaluar el Modelo2 (OAA) sobre la base de datos LYRL2004.**

CLASE	TP	FP	FN	TN	$\pi$	$\rho$	$F_1$
CCAT	343476	38420	27065	372304	0,899	0,927	0,913
C11	7923	9144	15728	748470	0,464	0,335	0,389
C12	7062	3252	4501	766450	0,685	0,611	0,646
C13	13074	12037	23389	732765	0,521	0,359	0,425
C14	2678	1503	4572	772512	0,641	0,369	0,469
C15	130381	17000	17225	616659	0,885	0,883	0,884
C151	69079	6855	10445	694886	0,910	0,869	0,889
C1511	17802	3072	5010	755381	0,853	0,780	0,815
C152	55440	16455	15722	693648	0,771	0,779	0,775
C16	342	57	1529	779337	0,857	0,183	0,301
C17	28501	11912	12482	728370	0,705	0,695	0,700
C171	11550	5732	6326	757657	0,668	0,646	0,657
C172	8361	3596	2841	766467	0,699	0,746	0,722
C173	1259	875	1301	777830	0,590	0,492	0,536
C174	4579	564	1046	775076	0,890	0,814	0,850
C18	37511	9960	13844	719950	0,790	0,730	0,759
C181	30784	10699	11385	728397	0,742	0,730	0,736
C182	619	367	3910	776369	0,628	0,137	0,224
C183	3567	1864	3637	772197	0,657	0,495	0,565
C21	10684	10627	13926	746028	0,501	0,434	0,465
C22	1621	2524	4308	772812	0,391	0,273	0,322
C23	692	793	1871	777909	0,466	0,270	0,342
C24	13289	7261	17942	742773	0,647	0,426	0,513
C31	20497	21769	18954	720045	0,485	0,520	0,502
C311	2078	5323	2055	771809	0,281	0,503	0,360
C312	2956	8807	3496	766006	0,251	0,458	0,325
C313	274	5258	800	774933	0,050	0,255	0,083
C32	191	61	1850	779163	0,758	0,094	0,167
C33	7393	6941	7496	759435	0,516	0,497	0,506
C331	235	61	944	780025	0,794	0,199	0,319
C34	1416	726	3299	775824	0,661	0,300	0,413
C41	8131	1566	2912	768656	0,839	0,736	0,784
C411	7986	1693	2000	769586	0,825	0,800	0,812
C42	7425	3477	4110	766253	0,681	0,644	0,662
ECAT	89746	22785	26725	642009	0,798	0,771	0,784
E11	3836	2428	4453	770548	0,612	0,463	0,527
E12	12243	8151	14178	746693	0,600	0,463	0,523
E121	1263	385	825	778792	0,766	0,605	0,676
E13	4595	1502	1821	773347	0,754	0,716	0,734
E131	3817	1663	1675	774110	0,697	0,695	0,696
E132	225	21	697	780322	0,915	0,244	0,385
E14	751	205	1361	778948	0,786	0,356	0,490
E141	60	37	304	780864	0,619	0,165	0,260
E142	16	1	176	781072	0,941	0,083	0,153
E143	540	134	632	779959	0,801	0,461	0,585
E21	29616	7547	12259	731843	0,797	0,707	0,749
E211	8700	4154	6661	761750	0,677	0,566	0,617
E212	19688	4259	6864	750454	0,822	0,741	0,780
E31	1214	271	1135	778645	0,818	0,517	0,633
E311	979	453	679	779154	0,684	0,590	0,634
E313	3	5	105	781152	0,375	0,028	0,052

Tabla C.2. Continuación							
CLASE	TP	FP	FN	TN	$\pi$	$\rho$	$F_1$
E41	11109	3097	5477	761582	0,782	0,670	0,722
E411	558	289	1538	778880	0,659	0,266	0,379
E51	10177	7487	10462	753139	0,576	0,493	0,531
E511	1095	1677	1736	776757	0,395	0,387	0,391
E512	6318	5529	5916	763502	0,533	0,516	0,525
E513	1185	190	1051	778839	0,862	0,530	0,656
E61	83	66	293	780823	0,557	0,221	0,316
E71	4405	206	697	775957	0,955	0,863	0,907
GCAT	209498	20705	22799	528263	0,910	0,902	0,906
G15	14355	3472	5954	757484	0,805	0,707	0,753
G151	272	372	2986	777635	0,422	0,083	0,139
G152	359	1755	1713	777438	0,170	0,173	0,172
G153	754	393	1547	778571	0,657	0,328	0,437
G154	5027	1264	3239	771735	0,799	0,608	0,691
G155	195	335	1891	778844	0,368	0,093	0,149
G156	0	0	258	781007	0,000	0,000	0,000
G157	240	37	1751	779237	0,866	0,121	0,212
G158	884	201	3364	776816	0,815	0,208	0,332
G159	0	0	38	781227	0,000	0,000	0,000
GCRIM	21791	10325	9295	739854	0,679	0,701	0,690
GDEF	3631	1878	4978	770778	0,659	0,422	0,514
GDIP	23477	13624	13258	730906	0,633	0,639	0,636
GDIS	5198	2291	3166	770610	0,694	0,621	0,656
GENT	1289	916	2406	776654	0,585	0,349	0,437
GENV	2362	1988	3727	773188	0,543	0,388	0,453
GFAS	0	0	307	780958	0,000	0,000	0,000
GHEA	3071	2456	2762	772976	0,556	0,526	0,541
GJOB	10369	3642	6401	760853	0,740	0,618	0,674
GOBIT	2	1	829	780433	0,667	0,002	0,005
GODD	257	839	2455	777714	0,234	0,095	0,135
GPOL	40076	29145	15155	696889	0,579	0,726	0,644
GPRO	1621	2170	3711	773763	0,428	0,304	0,355
GREL	1123	371	1634	778137	0,752	0,407	0,528
GSCI	647	103	1726	778789	0,863	0,273	0,414
GSPO	32751	1137	1653	745724	0,966	0,952	0,959
GTOUR	175	21	482	780587	0,893	0,266	0,410
GVIO	21478	11344	10022	738421	0,654	0,682	0,668
GVOTE	4845	2413	6341	767666	0,668	0,433	0,525
GWEA	1443	402	2300	777120	0,782	0,386	0,516
GWELF	418	135	1400	779312	0,756	0,230	0,353
MCAT	179621	18314	19317	564013	0,907	0,903	0,905
M11	41252	5896	6150	727967	0,875	0,870	0,873
M12	19275	5777	6029	750184	0,769	0,762	0,766
M13	43437	8047	8601	721180	0,844	0,835	0,839
M131	21738	4012	5504	750011	0,844	0,798	0,820
M132	20595	5788	5458	749424	0,781	0,791	0,786
M14	72765	6082	10134	692284	0,923	0,878	0,900
M141	41935	6251	4265	728814	0,870	0,908	0,889
M142	8932	1121	2887	768325	0,888	0,756	0,817
M143	18182	2567	3169	757347	0,876	0,852	0,864
Micro-media	1946418	480383	586703	75894261	0,802	0,768	0,785
Macro-media					0,668	0,502	0,544

**Tabla C.3. Valores obtenidos al evaluar el Modelo3 (BCE) sobre la base de datos LYRL2004.**

CLASE	TP	FP	FN	TN	$\pi$	$\rho$	$F_1$
CCAT	341517	23093	29024	387631	0,937	0,922	0,929
C11	8467	8629	15184	748985	0,495	0,358	0,416
C12	6431	2370	5132	767332	0,731	0,556	0,632
C13	13309	10387	23154	734415	0,562	0,365	0,442
C14	3979	2971	3271	771044	0,573	0,549	0,560
C15	128403	10871	19203	622788	0,922	0,870	0,895
C151	68910	5666	10614	696075	0,924	0,867	0,894
C1511	17794	2931	5018	755522	0,859	0,780	0,817
C152	53555	11323	17607	698780	0,825	0,753	0,787
C16	548	167	1323	779227	0,766	0,293	0,424
C17	27405	8936	13578	731346	0,754	0,669	0,709
C171	11309	5244	6567	758145	0,683	0,633	0,657
C172	8289	2933	2913	767130	0,739	0,740	0,739
C173	1084	604	1476	778101	0,642	0,423	0,510
C174	4563	570	1062	775070	0,889	0,811	0,848
C18	40177	11401	11178	718509	0,779	0,782	0,781
C181	33266	13059	8903	726037	0,718	0,789	0,752
C182	1281	1158	3248	775578	0,525	0,283	0,368
C183	3794	1825	3410	772236	0,675	0,527	0,592
C21	10565	6778	14045	749877	0,609	0,429	0,504
C22	2040	1702	3889	773634	0,545	0,344	0,422
C23	845	536	1718	778166	0,612	0,330	0,428
C24	13782	6525	17449	743509	0,679	0,441	0,535
C31	18177	9972	21274	731842	0,646	0,461	0,538
C311	2128	4118	2005	773014	0,341	0,515	0,410
C312	2818	7987	3634	766826	0,261	0,437	0,327
C313	198	3519	876	776672	0,053	0,184	0,083
C32	405	114	1636	779110	0,780	0,198	0,316
C33	6992	4118	7897	762258	0,629	0,470	0,538
C331	191	23	988	780063	0,893	0,162	0,274
C34	1686	877	3029	775673	0,658	0,358	0,463
C41	8084	1328	2959	768894	0,859	0,732	0,790
C411	7888	1497	2098	769782	0,840	0,790	0,814
C42	7589	2387	3946	767343	0,761	0,658	0,706
ECAT	91322	17114	25149	647680	0,842	0,784	0,812
E11	4127	2627	4162	770349	0,611	0,498	0,549
E12	14451	8936	11970	745908	0,618	0,547	0,580
E121	1342	410	746	778767	0,766	0,643	0,699
E13	4883	1542	1533	773307	0,760	0,761	0,761
E131	4146	1683	1346	774090	0,711	0,755	0,732
E132	208	17	714	780326	0,924	0,226	0,363
E14	949	403	1163	778750	0,702	0,449	0,548
E141	100	74	264	780827	0,575	0,275	0,372
E142	17	1	175	781072	0,944	0,089	0,162
E143	681	248	491	779845	0,733	0,581	0,648
E21	30629	5999	11246	733391	0,836	0,731	0,780
E211	9396	3501	5965	762403	0,729	0,612	0,665
E212	19883	3151	6669	751562	0,863	0,749	0,802
E31	1109	183	1240	778733	0,858	0,472	0,609
E311	881	359	777	779248	0,710	0,531	0,608
E313	2	7	106	781150	0,222	0,019	0,034

Tabla C.3. Continuación							
CLASE	TP	FP	FN	TN	$\pi$	$\rho$	F <sub>1</sub>
E41	11512	3310	5074	761369	0,777	0,694	0,733
E411	563	367	1533	778802	0,605	0,269	0,372
E51	10253	5421	10386	755205	0,654	0,497	0,565
E511	1200	1593	1631	776841	0,430	0,424	0,427
E512	6587	4570	5647	764461	0,590	0,538	0,563
E513	1072	138	1164	778891	0,886	0,479	0,622
E61	24	25	352	780864	0,490	0,064	0,113
E71	4694	213	408	775950	0,957	0,920	0,938
GCAT	213807	20315	18490	528653	0,913	0,920	0,917
G15	15304	3425	5005	757531	0,817	0,754	0,784
G151	415	590	2843	777417	0,413	0,127	0,195
G152	141	474	1931	778719	0,229	0,068	0,105
G153	913	556	1388	778408	0,622	0,397	0,484
G154	5991	1555	2275	771444	0,794	0,725	0,758
G155	185	272	1901	778907	0,405	0,089	0,145
G156	0	0	258	781007	0,000	0,000	0,000
G157	562	128	1429	779146	0,814	0,282	0,419
G158	1957	1205	2291	775812	0,619	0,461	0,528
G159	0	0	38	781227	0,000	0,000	0,000
GCRIM	22311	10095	8775	740084	0,688	0,718	0,703
GDEF	4106	3170	4503	769486	0,564	0,477	0,517
GDIP	21674	10029	15061	734501	0,684	0,590	0,633
GDIS	4866	1643	3498	771258	0,748	0,582	0,654
GENT	1583	1005	2112	776565	0,612	0,428	0,504
GENV	1852	1461	4237	773715	0,559	0,304	0,394
GFAS	0	0	307	780958	0,000	0,000	0,000
GHEA	2956	2157	2877	773275	0,578	0,507	0,540
GJOB	11577	4313	5193	760182	0,729	0,690	0,709
GOBIT	3	2	828	780432	0,600	0,004	0,007
GODD	213	687	2499	777866	0,237	0,079	0,118
GPOL	37321	22379	17910	703655	0,625	0,676	0,649
GPRO	1121	1003	4211	774930	0,528	0,210	0,301
GREL	996	384	1761	778124	0,722	0,361	0,482
GSCI	1030	165	1343	778727	0,862	0,434	0,577
GSPO	32959	1358	1445	745503	0,960	0,958	0,959
GTOUR	136	17	521	780591	0,889	0,207	0,336
GVIO	20215	8539	11285	741226	0,703	0,642	0,671
GVOTE	6059	3080	5127	766999	0,663	0,542	0,596
GWEA	1877	558	1866	776964	0,771	0,501	0,608
GWELF	366	196	1452	779251	0,651	0,201	0,308
MCAT	182336	15168	16602	567159	0,923	0,917	0,920
M11	41991	4642	5411	729221	0,900	0,886	0,893
M12	20446	5434	4858	750527	0,790	0,808	0,799
M13	44889	7209	7149	722018	0,862	0,863	0,862
M131	22463	3571	4779	750452	0,863	0,825	0,843
M132	21074	5155	4979	750057	0,803	0,809	0,806
M14	74854	5526	8045	692840	0,931	0,903	0,917
M141	42945	6651	3255	728414	0,866	0,930	0,897
M142	9414	1268	2405	768178	0,881	0,797	0,837
M143	18640	2609	2711	757305	0,877	0,873	0,875
Micro-media	1969048	395505	564073	75979139	0,833	0,777	0,804
Macro-media					0,680	0,521	0,566

**Tabla C.4. Valores obtenidos al evaluar el Modelo4 (CCE) sobre la base de datos LYRL2004.**

CLASE	TP	FP	FN	TN	$\pi$	$\rho$	$F_1$
CCAT	364905	260056	5636	150668	0,584	0,985	0,733
C11	13020	24392	10631	733222	0,348	0,551	0,426
C12	8681	15925	2882	753777	0,353	0,751	0,480
C13	21736	51998	14727	692804	0,295	0,596	0,394
C14	5311	8447	1939	765568	0,386	0,733	0,506
C15	136023	60164	11583	573495	0,693	0,922	0,791
C151	70116	9335	9408	692406	0,883	0,882	0,882
C1511	17879	18438	4933	740015	0,492	0,784	0,605
C152	58892	53451	12270	656652	0,524	0,828	0,642
C16	584	184	1287	779210	0,760	0,312	0,443
C17	32440	24208	8543	716074	0,573	0,792	0,665
C171	13973	16850	3903	746539	0,453	0,782	0,574
C172	10055	24250	1147	745813	0,293	0,898	0,442
C173	1791	2705	769	776000	0,398	0,700	0,508
C174	4961	7640	664	768000	0,394	0,882	0,544
C18	44450	22910	6905	707000	0,660	0,866	0,749
C181	36959	25642	5210	713454	0,590	0,876	0,706
C182	3355	33558	1174	743178	0,091	0,741	0,162
C183	3518	1465	3686	772596	0,706	0,488	0,577
C21	13538	21505	11072	735150	0,386	0,550	0,454
C22	2616	3327	3313	772009	0,440	0,441	0,441
C23	1247	1856	1316	776846	0,402	0,487	0,440
C24	19886	23695	11345	726339	0,456	0,637	0,532
C31	26959	52411	12492	689403	0,340	0,683	0,454
C311	1449	966	2684	776166	0,600	0,351	0,443
C312	3858	7741	2594	767072	0,333	0,598	0,427
C313	5	8	1069	780183	0,385	0,005	0,009
C32	772	280	1269	778944	0,734	0,378	0,499
C33	8583	14960	6306	751416	0,365	0,576	0,447
C331	283	57	896	780029	0,832	0,240	0,373
C34	2336	1538	2379	775012	0,603	0,495	0,544
C41	8942	4691	2101	765531	0,656	0,810	0,725
C411	8645	4869	1341	766410	0,640	0,866	0,736
C42	8880	5825	2655	763905	0,604	0,770	0,677
ECAT	106225	180117	10246	484677	0,371	0,912	0,527
E11	6016	13528	2273	759448	0,308	0,726	0,432
E12	20275	71678	6146	683166	0,220	0,767	0,343
E121	1485	824	603	778353	0,643	0,711	0,675
E13	5395	5806	1021	769043	0,482	0,841	0,612
E131	4566	5679	926	770094	0,446	0,831	0,580
E132	215	23	707	780320	0,903	0,233	0,371
E14	1161	1604	951	777549	0,420	0,550	0,476
E141	30	7	334	780894	0,811	0,082	0,150
E142	23	5	169	781068	0,821	0,120	0,209
E143	372	38	800	780055	0,907	0,317	0,470
E21	37631	62798	4244	676592	0,375	0,899	0,529
E211	12165	22762	3196	743142	0,348	0,792	0,484
E212	23164	28163	3388	726550	0,451	0,872	0,595
E31	1498	659	851	778257	0,694	0,638	0,665
E311	1177	853	481	778754	0,580	0,710	0,638
E313	2	9	106	781148	0,182	0,019	0,034

Tabla C.4. Continuación							
CLASE	TP	FP	FN	TN	$\pi$	$\rho$	F <sub>1</sub>
E41	13336	13227	3250	751452	0,502	0,804	0,618
E411	574	380	1522	778789	0,602	0,274	0,376
E51	14653	39351	5986	721275	0,271	0,710	0,393
E511	716	416	2115	778018	0,633	0,253	0,361
E512	7688	9582	4546	759449	0,445	0,628	0,521
E513	1086	54	1150	778975	0,953	0,486	0,643
E61	121	87	255	780802	0,582	0,322	0,414
E71	4804	682	298	775481	0,876	0,942	0,907
GCAT	225444	154039	6853	394929	0,594	0,970	0,737
G15	17552	6809	2757	754147	0,720	0,864	0,786
G151	1766	2900	1492	775107	0,378	0,542	0,446
G152	354	1643	1718	777550	0,177	0,171	0,174
G153	1345	870	956	778094	0,607	0,585	0,596
G154	6758	3768	1508	769231	0,642	0,818	0,719
G155	981	3341	1105	775838	0,227	0,470	0,306
G156	0	0	258	781007	0,000	0,000	0,000
G157	931	238	1060	779036	0,796	0,468	0,589
G158	2607	2481	1641	774536	0,512	0,614	0,558
G159	0	0	38	781227	0,000	0,000	0,000
GCRIM	25844	21388	5242	728791	0,547	0,831	0,660
GDEF	4536	3657	4073	768999	0,554	0,527	0,540
GDIP	26847	18363	9888	726167	0,594	0,731	0,655
GDIS	6197	4261	2167	768640	0,593	0,741	0,658
GENT	2118	2508	1577	775062	0,458	0,573	0,509
GENV	3123	5098	2966	770078	0,380	0,513	0,436
GFAS	0	0	307	780958	0,000	0,000	0,000
GHEA	4053	5170	1780	770262	0,439	0,695	0,538
GJOB	13482	12124	3288	752371	0,527	0,804	0,636
GOBIT	20	11	811	780423	0,645	0,024	0,046
GODD	487	2092	2225	776461	0,189	0,180	0,184
GPOL	46320	49765	8911	676269	0,482	0,839	0,612
GPRO	1935	2300	3397	773633	0,457	0,363	0,405
GREL	1561	735	1196	777773	0,680	0,566	0,618
GSCI	1327	498	1046	778394	0,727	0,559	0,632
GSPO	33357	2205	1047	744656	0,938	0,970	0,954
GTOUR	346	185	311	780423	0,652	0,527	0,582
GVIO	24234	12655	7266	737110	0,657	0,769	0,709
GVOTE	8765	6613	2421	763466	0,570	0,784	0,660
GWEA	2390	1098	1353	776424	0,685	0,639	0,661
GWELF	679	563	1139	778884	0,547	0,373	0,444
MCAT	192578	115306	6360	467021	0,625	0,968	0,760
M11	45934	102598	1468	631265	0,309	0,969	0,469
M12	23762	61731	1542	694230	0,278	0,939	0,429
M13	49685	67240	2353	661987	0,425	0,955	0,588
M131	24631	22177	2611	731846	0,526	0,904	0,665
M132	23159	28204	2894	727008	0,451	0,889	0,598
M14	79958	62681	2941	635685	0,561	0,965	0,709
M141	41124	3970	5076	731095	0,912	0,890	0,901
M142	10299	2606	1520	766840	0,798	0,871	0,833
M143	18270	1835	3081	758079	0,909	0,856	0,881
Micro-media	2195785	2059405	337336	74315239	0,516	0,867	0,647
Macro-media					0,523	0,628	0,524

**Tabla C.5. Valores obtenidos al evaluar *knn* optimizado para micro-valores sobre la base de datos LYRL2004 [Lewis et. al, 2004].**

CLASE	TP	FP	FN	TN	$\pi$	$\rho$	F <sub>1</sub>
CCAT	341903	38337	28638	372387	0,899	0,923	0,911
C11	487	34	23164	757580	0,935	0,021	0,040
C12	6397	4128	5166	765574	0,608	0,553	0,579
C13	993	243	35470	744559	0,803	0,027	0,053
C14	4628	7170	2622	766845	0,392	0,638	0,486
C15	121706	11387	25900	622272	0,914	0,825	0,867
C151	63861	5587	15663	696154	0,920	0,803	0,857
C1511	11970	5408	10842	753045	0,689	0,525	0,596
C152	55146	33182	16016	676921	0,624	0,775	0,692
C16	874	627	997	778767	0,582	0,467	0,518
C17	27591	11587	13392	728695	0,704	0,673	0,688
C171	10991	6620	6885	756769	0,624	0,615	0,619
C172	7337	2400	3865	767663	0,754	0,655	0,701
C173	1587	1841	973	776864	0,463	0,620	0,530
C174	4866	1485	759	774155	0,766	0,865	0,813
C18	38351	17455	13004	712455	0,687	0,747	0,716
C181	31437	16017	10732	723079	0,662	0,746	0,702
C182	213	263	4316	776473	0,447	0,047	0,085
C183	3732	4035	3472	770026	0,480	0,518	0,499
C21	4176	1061	20434	755594	0,797	0,170	0,280
C22	287	303	5642	775033	0,486	0,048	0,088
C23	769	675	1794	778027	0,533	0,300	0,384
C24	18035	17879	13196	732155	0,502	0,577	0,537
C31	568	27	38883	741787	0,955	0,014	0,028
C311	435	73	3698	777059	0,856	0,105	0,187
C312	226	144	6226	774669	0,611	0,035	0,066
C313	4	32	1070	780159	0,111	0,004	0,007
C32	276	71	1765	779153	0,795	0,135	0,231
C33	2637	663	12252	765713	0,799	0,177	0,290
C331	166	46	1013	780040	0,783	0,141	0,239
C34	2204	3782	2511	772768	0,368	0,467	0,412
C41	7697	1544	3346	768678	0,833	0,697	0,759
C411	7459	1643	2527	769636	0,819	0,747	0,782
C42	7271	2651	4264	767079	0,733	0,630	0,678
ECAT	92849	32199	23622	632595	0,743	0,797	0,769
E11	4647	3628	3642	769348	0,562	0,561	0,561
E12	16088	16310	10333	738534	0,497	0,609	0,547
E121	1695	1797	393	777380	0,485	0,812	0,608
E13	5240	3997	1176	770852	0,567	0,817	0,670
E131	4086	2399	1406	773374	0,630	0,744	0,682
E132	383	460	539	779883	0,454	0,415	0,434
E14	994	799	1118	778354	0,554	0,471	0,509
E141	90	85	274	780816	0,514	0,247	0,334
E142	70	484	122	780589	0,126	0,365	0,188
E143	758	705	414	779388	0,518	0,647	0,575
E21	32342	12594	9533	726796	0,720	0,772	0,745
E211	10204	6984	5157	758920	0,594	0,664	0,627
E212	18836	3955	7716	750758	0,826	0,709	0,763
E31	1320	574	1029	778342	0,697	0,562	0,622
E311	1072	749	586	778858	0,589	0,647	0,616
E313	8	31	100	781126	0,205	0,074	0,109



Tabla C.5. Continuación							
CLASE	TP	FP	FN	TN	$\pi$	$\rho$	$F_1$
E41	11028	5203	5558	759476	0,679	0,665	0,672
E411	1496	1132	600	778037	0,569	0,714	0,633
E51	10814	8059	9825	752567	0,573	0,524	0,547
E511	705	453	2126	777981	0,609	0,249	0,353
E512	7184	5862	5050	763169	0,551	0,587	0,568
E513	1354	459	882	778570	0,747	0,606	0,669
E61	92	125	284	780764	0,424	0,245	0,310
E71	4687	654	415	775509	0,878	0,919	0,898
GCAT	214702	25684	17595	523284	0,893	0,924	0,908
G15	16310	6162	3999	754794	0,726	0,803	0,762
G151	2119	3666	1139	774341	0,366	0,650	0,469
G152	93	254	1979	778939	0,268	0,045	0,077
G153	1829	2706	472	776258	0,403	0,795	0,535
G154	7031	2782	1235	770217	0,716	0,851	0,778
G155	72	301	2014	778878	0,193	0,035	0,059
G156	0	58	258	780949	0,000	0,000	0,000
G157	255	63	1736	779211	0,802	0,128	0,221
G158	778	388	3470	776629	0,667	0,183	0,287
G159	0	0	38	781227	0,000	0,000	0,000
GCRIM	22397	8478	8689	741701	0,725	0,720	0,723
GDEF	3957	3105	4652	769551	0,560	0,460	0,505
GDIP	26636	18682	10099	725848	0,588	0,725	0,649
GDIS	6059	3453	2305	769448	0,637	0,724	0,678
GENT	2137	2292	1558	775278	0,483	0,578	0,526
GENV	284	238	5805	774938	0,544	0,047	0,086
GFAS	4	131	303	780827	0,030	0,013	0,018
GHEA	4408	4796	1425	770636	0,479	0,756	0,586
GMIL	11807	6886	4963	757609	0,632	0,704	0,666
GOBIT	2	13	829	780421	0,133	0,002	0,005
GODD	3	3	2709	778550	0,500	0,001	0,002
GPOL	38072	19515	17159	706519	0,661	0,689	0,675
GPRO	1305	1644	4027	774289	0,443	0,245	0,315
GREL	225	67	2532	778441	0,771	0,082	0,148
GSCI	1820	3142	553	775750	0,367	0,767	0,496
GSPO	33690	1249	714	745612	0,964	0,979	0,972
GTOUR	352	264	305	780344	0,571	0,536	0,553
GVIO	22468	12362	9032	737403	0,645	0,713	0,677
GVOTE	5522	4647	5664	765432	0,543	0,494	0,517
GWEA	2837	1626	906	775896	0,636	0,758	0,691
GWELF	208	47	1610	779400	0,816	0,114	0,201
MCAT	185693	30602	13245	551725	0,859	0,933	0,894
M11	40483	6202	6919	727661	0,867	0,854	0,861
M12	20619	8138	4685	747823	0,717	0,815	0,763
M13	44186	11783	7852	717444	0,789	0,849	0,818
M131	22531	6652	4711	747371	0,772	0,827	0,799
M132	21206	8926	4847	746286	0,704	0,814	0,755
M14	78434	12817	4465	685549	0,860	0,946	0,901
M141	44286	7457	1914	727608	0,856	0,959	0,904
M142	10383	3496	1436	765950	0,748	0,879	0,808
M143	18450	2341	2901	757573	0,887	0,864	0,876
Micro-media	1927975	541215	605146	75833429	0,781	0,761	0,771
Macro-media					0,615	0,524	0,516

**Tabla C.6. Valores obtenidos al evaluar *knn* optimizado para macro-valores sobre la base de datos LYRL2004 [Lewis et. al, 2004].**

CLASE	TP	FP	FN	TN	$\pi$	$\rho$	$F_1$
CCAT	343978	41257	26563	369467	0,893	0,928	0,910
C11	11878	26174	11773	731440	0,312	0,502	0,385
C12	6571	4459	4992	765243	0,596	0,568	0,582
C13	19213	31657	17250	713145	0,378	0,527	0,440
C14	4254	5879	2996	768136	0,420	0,587	0,489
C15	125902	16456	21704	617203	0,884	0,853	0,868
C151	64580	6260	14944	695481	0,912	0,812	0,859
C1511	13346	10696	9466	747757	0,555	0,585	0,570
C152	49896	22875	21266	687228	0,686	0,701	0,693
C16	704	295	1167	779099	0,705	0,376	0,491
C17	27840	12168	13143	728114	0,696	0,679	0,687
C171	11813	9145	6063	754244	0,564	0,661	0,608
C172	7364	2431	3838	767632	0,752	0,657	0,701
C173	1606	1916	954	776789	0,456	0,627	0,528
C174	4781	1175	844	774465	0,803	0,850	0,826
C18	37266	15219	14089	714691	0,710	0,726	0,718
C181	32593	18960	9576	720136	0,632	0,773	0,696
C182	1397	4840	3132	771896	0,224	0,308	0,260
C183	4183	5436	3021	768625	0,435	0,581	0,497
C21	11913	14445	12697	742210	0,452	0,484	0,467
C22	3433	10569	2496	764767	0,245	0,579	0,344
C23	989	1747	1574	776955	0,361	0,386	0,373
C24	18940	20390	12291	729644	0,482	0,606	0,537
C31	21227	30997	18224	710817	0,406	0,538	0,463
C311	1598	2385	2535	774747	0,401	0,387	0,394
C312	2674	5697	3778	769116	0,319	0,414	0,361
C313	20	214	1054	779977	0,085	0,019	0,031
C32	914	478	1127	778746	0,657	0,448	0,532
C33	8462	15344	6427	751032	0,355	0,568	0,437
C331	682	1170	497	778916	0,368	0,578	0,450
C34	2347	4508	2368	772042	0,342	0,498	0,406
C41	7483	1301	3560	768921	0,852	0,678	0,755
C411	7541	1784	2445	769495	0,809	0,755	0,781
C42	7581	3179	3954	766551	0,705	0,657	0,680
ECAT	92288	30782	24183	634012	0,750	0,792	0,771
E11	5064	5235	3225	767741	0,492	0,611	0,545
E12	16740	18785	9681	736059	0,471	0,634	0,540
E121	1558	880	530	778297	0,639	0,746	0,688
E13	4899	2803	1517	772046	0,636	0,764	0,694
E131	4253	2959	1239	772814	0,590	0,774	0,670
E132	543	1308	379	779035	0,293	0,589	0,392
E14	1141	1532	971	777621	0,427	0,540	0,477
E141	149	661	215	780240	0,184	0,409	0,254
E142	91	972	101	780101	0,086	0,474	0,145
E143	744	603	428	779490	0,552	0,635	0,591
E21	31401	10899	10474	728491	0,742	0,750	0,746
E211	9891	6308	5470	759596	0,611	0,644	0,627
E212	19696	5129	6856	749584	0,793	0,742	0,767
E31	1325	576	1024	778340	0,697	0,564	0,624
E311	963	453	695	779154	0,680	0,581	0,627
E313	68	562	40	780595	0,108	0,630	0,184

Tabla C.6. Continuación							
CLASE	TP	FP	FN	TN	$\pi$	$\rho$	$F_1$
E41	11236	5857	5350	758822	0,657	0,677	0,667
E411	1428	994	668	778175	0,590	0,681	0,632
E51	11669	10241	8970	750385	0,533	0,565	0,548
E511	952	1029	1879	777405	0,481	0,336	0,396
E512	7586	7004	4648	762027	0,520	0,620	0,566
E513	1395	661	841	778368	0,679	0,624	0,650
E61	144	220	232	780669	0,396	0,383	0,389
E71	4734	963	368	775200	0,831	0,928	0,877
GCAT	214773	25775	17524	523193	0,893	0,925	0,908
G15	16342	6245	3967	754711	0,724	0,805	0,762
G151	2622	5813	636	772194	0,311	0,805	0,448
G152	79	199	1993	778994	0,284	0,038	0,067
G153	1854	2841	447	776123	0,395	0,806	0,530
G154	7039	2810	1227	770189	0,715	0,852	0,777
G155	916	4587	1170	774592	0,166	0,439	0,241
G156	0	13	258	780994	0,000	0,000	0,000
G157	457	328	1534	778946	0,582	0,230	0,329
G158	2266	3110	1982	773907	0,422	0,533	0,471
G159	0	0	38	781227	0,000	0,000	0,000
GCRIM	21936	7622	9150	742557	0,742	0,706	0,723
GDEF	4238	3926	4371	768730	0,519	0,492	0,505
GDIP	27106	19647	9629	724883	0,580	0,738	0,649
GDIS	6012	3387	2352	769514	0,640	0,719	0,677
GENT	2085	2088	1610	775482	0,500	0,564	0,530
GENV	3152	3901	2937	771275	0,447	0,518	0,480
GFAS	4	131	303	780827	0,030	0,013	0,018
GHEA	4384	4701	1449	770731	0,483	0,752	0,588
GMIL	11493	5686	5277	758809	0,669	0,685	0,677
GOBIT	71	399	760	780035	0,151	0,085	0,109
GODD	477	3591	2235	774962	0,117	0,176	0,141
GPOL	39907	22706	15324	703328	0,637	0,723	0,677
GPRO	1262	1402	4070	774531	0,474	0,237	0,316
GREL	1111	1339	1646	777169	0,453	0,403	0,427
GSCI	1791	2775	582	776117	0,392	0,755	0,516
GSPO	33815	1471	589	745390	0,958	0,983	0,970
GTOUR	357	321	300	780287	0,527	0,543	0,535
GVIO	22280	12106	9220	737659	0,648	0,707	0,676
GVOTE	5891	5236	5295	764843	0,529	0,527	0,528
GWEA	2684	1165	1059	776357	0,697	0,717	0,707
GWELF	616	1187	1202	778260	0,342	0,339	0,340
MCAT	186934	33242	12004	549085	0,849	0,940	0,892
M11	40531	6275	6871	727588	0,866	0,855	0,860
M12	21228	10057	4076	745904	0,679	0,839	0,750
M13	43667	10562	8371	718665	0,805	0,839	0,822
M131	22201	5797	5041	748226	0,793	0,815	0,804
M132	21348	9407	4705	745805	0,694	0,819	0,752
M14	78050	11866	4849	686500	0,868	0,942	0,903
M141	44540	8223	1660	726842	0,844	0,964	0,900
M142	10406	3560	1413	765886	0,745	0,880	0,807
M143	18367	2231	2984	757683	0,892	0,860	0,876
Micro-media	2019219	726650	513902	75647994	0,735	0,797	0,765
Macro-media					0,544	0,606	0,560

**Tabla C.7. Valores obtenidos al evaluar *Rocchio* sobre la base de datos LYRL2004 [Lewis et. al, 2004].**

CLASE	TP	FP	FN	TN	$\pi$	$\rho$	$F_1$
CCAT	316945	101792	53596	308932	0,757	0,855	0,803
C11	9336	17313	14315	740301	0,350	0,395	0,371
C12	5780	4766	5783	764936	0,548	0,500	0,523
C13	14895	31469	21568	713333	0,321	0,408	0,360
C14	3713	4301	3537	769714	0,463	0,512	0,487
C15	112751	22040	34855	611619	0,836	0,764	0,799
C151	58190	8113	21334	693628	0,878	0,732	0,798
C1511	5783	7882	17029	750571	0,423	0,254	0,317
C152	43947	33608	27215	676495	0,567	0,618	0,591
C16	763	384	1108	779010	0,665	0,408	0,506
C17	23770	15911	17213	724371	0,599	0,580	0,589
C171	9532	5608	8344	757781	0,630	0,533	0,577
C172	5146	476	6056	769587	0,915	0,459	0,612
C173	879	541	1681	778164	0,619	0,343	0,442
C174	2947	432	2678	775208	0,872	0,524	0,655
C18	34624	21265	16731	708645	0,620	0,674	0,646
C181	29347	20421	12822	718675	0,590	0,696	0,638
C182	1554	7835	2975	768901	0,166	0,343	0,223
C183	4112	2851	3092	771210	0,591	0,571	0,581
C21	8289	11489	16321	745166	0,419	0,337	0,373
C22	1658	4183	4271	771153	0,284	0,280	0,282
C23	1056	1802	1507	776900	0,369	0,412	0,390
C24	13338	14865	17893	735169	0,473	0,427	0,449
C31	13890	27352	25561	714462	0,337	0,352	0,344
C311	963	760	3170	776372	0,559	0,233	0,329
C312	2724	5805	3728	769008	0,319	0,422	0,364
C313	8	31	1066	780160	0,205	0,007	0,014
C32	784	281	1257	778943	0,736	0,384	0,505
C33	6317	7180	8572	759196	0,468	0,424	0,445
C331	404	319	775	779767	0,559	0,343	0,425
C34	2103	1518	2612	775032	0,581	0,446	0,505
C41	6560	1617	4483	768605	0,802	0,594	0,683
C411	6443	1754	3543	769525	0,786	0,645	0,709
C42	6795	2251	4740	767479	0,751	0,589	0,660
ECAT	72124	43945	44347	620849	0,621	0,619	0,620
E11	3398	2202	4891	770774	0,607	0,410	0,489
E12	11461	13133	14960	741711	0,466	0,434	0,449
E121	1023	276	1065	778901	0,788	0,490	0,604
E13	3866	1424	2550	773425	0,731	0,603	0,661
E131	3786	1825	1706	773948	0,675	0,689	0,682
E132	155	110	767	780233	0,585	0,168	0,261
E14	552	173	1560	778980	0,761	0,261	0,389
E141	20	1	344	780900	0,952	0,055	0,104
E142	30	8	162	781065	0,789	0,156	0,261
E143	483	131	689	779962	0,787	0,412	0,541
E21	27107	16696	14768	722694	0,619	0,647	0,633
E211	6996	2735	8365	763169	0,719	0,455	0,558
E212	16649	6168	9903	748545	0,730	0,627	0,674
E31	985	292	1364	778624	0,771	0,419	0,543
E311	738	224	920	779383	0,767	0,445	0,563
E313	6	13	102	781144	0,316	0,056	0,094

Tabla C.7. Continuación							
CLASE	TP	FP	FN	TN	$\pi$	$\rho$	$F_1$
E41	10088	3139	6498	761540	0,763	0,608	0,677
E411	1026	526	1070	778643	0,661	0,490	0,563
E51	7705	6962	12934	753664	0,525	0,373	0,436
E511	710	567	2121	777867	0,556	0,251	0,346
E512	5297	3650	6937	765381	0,592	0,433	0,500
E513	1019	64	1217	778965	0,941	0,456	0,614
E61	22	25	354	780864	0,468	0,059	0,104
E71	3966	197	1136	775966	0,953	0,777	0,856
GCAT	192683	43713	39614	505255	0,815	0,829	0,822
G15	16224	5218	4085	755738	0,757	0,799	0,777
G151	1473	2388	1785	775619	0,382	0,452	0,414
G152	118	189	1954	779004	0,384	0,057	0,099
G153	948	1045	1353	777919	0,476	0,412	0,442
G154	6100	2086	2166	770913	0,745	0,738	0,742
G155	215	634	1871	778545	0,253	0,103	0,147
G156	0	0	258	781007	0,000	0,000	0,000
G157	1116	676	875	778598	0,623	0,561	0,590
G158	1801	2241	2447	774776	0,446	0,424	0,434
G159	0	0	38	781227	0,000	0,000	0,000
GCRIM	21637	11191	9449	738988	0,659	0,696	0,677
GDEF	2808	2445	5801	770211	0,535	0,326	0,405
GDIP	21646	15889	15089	728641	0,577	0,589	0,583
GDIS	4343	2048	4021	770853	0,680	0,519	0,589
GENT	1576	1326	2119	776244	0,543	0,427	0,478
GENV	1529	1428	4560	773748	0,517	0,251	0,338
GFAS	0	0	307	780958	0,000	0,000	0,000
GHEA	2982	1862	2851	773570	0,616	0,511	0,559
GMIL	11211	4423	5559	760072	0,717	0,669	0,692
GOBIT	2	0	829	780434	1,000	0,002	0,005
GODD	0	0	2712	778553	0,000	0,000	0,000
GPOL	38007	28246	17224	697788	0,574	0,688	0,626
GPRO	637	375	4695	775558	0,629	0,119	0,201
GREL	449	99	2308	778409	0,819	0,163	0,272
GSCI	1337	525	1036	778367	0,718	0,563	0,631
GSPO	32520	1911	1884	744950	0,944	0,945	0,945
GTOUR	250	27	407	780581	0,903	0,381	0,535
GVIO	21164	10568	10336	739197	0,667	0,672	0,669
GVOTE	7042	4479	4144	765600	0,611	0,630	0,620
GWEA	2550	1321	1193	776201	0,659	0,681	0,670
GWELF	95	14	1723	779433	0,872	0,052	0,099
MCAT	150940	38286	47998	544041	0,798	0,759	0,778
M11	36024	4304	11378	729559	0,893	0,760	0,821
M12	16703	10484	8601	745477	0,614	0,660	0,636
M13	39956	11733	12082	717494	0,773	0,768	0,770
M131	19640	6598	7602	747425	0,749	0,721	0,734
M132	17676	4429	8377	750783	0,800	0,678	0,734
M14	68465	15749	14434	682617	0,813	0,826	0,819
M141	37391	6463	8809	728602	0,853	0,809	0,830
M142	9297	2405	2522	767041	0,794	0,787	0,791
M143	16619	5682	4732	754232	0,745	0,778	0,761
Micro-media	1739732	745201	793389	75629443	0,700	0,687	0,693
Macro-media					0,615	0,468	0,504

**Tabla C.8. Valores obtenidos al evaluar *svm1* optimizado para micro-medidas sobre la base de datos LYRL2004 [Lewis et. al, 2004].**

CLASE	TP	FP	FN	TN	$\pi$	$\rho$	$F_1$
CCAT	341459	24162	29082	386562	0,934	0,922	0,928
C11	101	5	23550	757609	0,953	0,004	0,009
C12	7442	4109	4121	765593	0,644	0,644	0,644
C13	19192	19930	17271	724872	0,491	0,526	0,508
C14	4898	4498	2352	769517	0,521	0,676	0,588
C15	125436	6085	22170	627574	0,954	0,850	0,899
C151	67422	3421	12102	698320	0,952	0,848	0,897
C1511	16460	4609	6352	753844	0,781	0,722	0,750
C152	53692	11950	17470	698153	0,818	0,755	0,785
C16	859	412	1012	778982	0,676	0,459	0,547
C17	27853	6052	13130	734230	0,822	0,680	0,744
C171	12661	4290	5215	759099	0,747	0,708	0,727
C172	8403	2128	2799	767935	0,798	0,750	0,773
C173	1557	1405	1003	777300	0,526	0,608	0,564
C174	5190	562	435	775078	0,902	0,923	0,912
C18	42276	13835	9079	716075	0,753	0,823	0,787
C181	32182	10368	9987	728728	0,756	0,763	0,760
C182	6	4	4523	776732	0,600	0,001	0,003
C183	4390	2275	2814	771786	0,659	0,609	0,633
C21	11724	8955	12886	747700	0,567	0,476	0,518
C22	220	69	5709	775267	0,761	0,037	0,071
C23	239	49	2324	778653	0,830	0,093	0,168
C24	17010	10228	14221	739806	0,624	0,545	0,582
C31	20310	15766	19141	726048	0,563	0,515	0,538
C311	2389	2331	1744	774801	0,506	0,578	0,540
C312	48	7	6404	774806	0,873	0,007	0,015
C313	3	4	1071	780187	0,429	0,003	0,006
C32	0	0	2041	779224	0,000	0,000	0,000
C33	8294	6513	6595	759863	0,560	0,557	0,559
C331	1	1	1178	780085	0,500	0,001	0,002
C34	2712	2649	2003	773901	0,506	0,575	0,538
C41	8504	1931	2539	768291	0,815	0,770	0,792
C411	8215	1936	1771	769343	0,809	0,823	0,816
C42	7975	2624	3560	767106	0,752	0,691	0,721
ECAT	89689	18365	26782	646429	0,830	0,770	0,799
E11	4564	2804	3725	770172	0,619	0,551	0,583
E12	14659	8614	11762	746230	0,630	0,555	0,590
E121	1497	359	591	778818	0,807	0,717	0,759
E13	4956	1744	1460	773105	0,740	0,772	0,756
E131	4348	2331	1144	773442	0,651	0,792	0,714
E132	602	501	320	779842	0,546	0,653	0,595
E14	1206	1142	906	778011	0,514	0,571	0,541
E141	82	54	282	780847	0,603	0,225	0,328
E142	69	58	123	781015	0,543	0,359	0,433
E143	806	572	366	779521	0,585	0,688	0,632
E21	31524	5939	10351	733451	0,841	0,753	0,795
E211	10708	4552	4653	761352	0,702	0,697	0,699
E212	20746	3327	5806	751386	0,862	0,781	0,820
E31	1417	438	932	778478	0,764	0,603	0,674
E311	1050	507	608	779100	0,674	0,633	0,653
E313	43	26	65	781131	0,623	0,398	0,486

Tabla C.8. Continuación							
CLASE	TP	FP	FN	TN	$\pi$	$\rho$	$F_1$
E41	12069	2906	4517	761773	0,806	0,728	0,765
E411	619	282	1477	778887	0,687	0,295	0,413
E51	12589	7573	8050	753053	0,624	0,610	0,617
E511	370	109	2461	778325	0,772	0,131	0,224
E512	7958	4443	4276	764588	0,642	0,650	0,646
E513	1339	131	897	778898	0,911	0,599	0,723
E61	129	117	247	780772	0,524	0,343	0,415
E71	4695	67	407	776096	0,986	0,920	0,952
GCAT	214239	15510	18058	533458	0,932	0,922	0,927
G15	17657	3845	2652	757111	0,821	0,869	0,845
G151	2	0	3256	778007	1,000	0,001	0,001
G152	0	0	2072	779193	0,000	0,000	0,000
G153	1609	1462	692	777502	0,524	0,699	0,599
G154	6115	1295	2151	771704	0,825	0,740	0,780
G155	0	0	2086	779179	0,000	0,000	0,000
G156	0	0	258	781007	0,000	0,000	0,000
G157	963	246	1028	779028	0,797	0,484	0,602
G158	2444	2344	1804	774673	0,510	0,575	0,541
G159	0	0	38	781227	0,000	0,000	0,000
GCRIM	23616	5698	7470	744481	0,806	0,760	0,782
GDEF	5034	2380	3575	770276	0,679	0,585	0,628
GDIP	26011	9107	10724	735423	0,741	0,708	0,724
GDIS	6373	2243	1991	770658	0,740	0,762	0,751
GENT	2112	1454	1583	776116	0,592	0,572	0,582
GENV	4386	4708	1703	770468	0,482	0,720	0,578
GFAS	1	3	306	780955	0,250	0,003	0,006
GHEA	3633	2126	2200	773306	0,631	0,623	0,627
GMIL	12171	2633	4599	761862	0,822	0,726	0,771
GOBIT	2	2	829	780432	0,500	0,002	0,005
GODD	1	0	2711	778553	1,000	0,000	0,001
GPOL	39779	15097	15452	710937	0,725	0,720	0,723
GPRO	962	486	4370	775447	0,664	0,180	0,284
GREL	315	6	2442	778502	0,981	0,114	0,205
GSCI	1702	1143	671	777749	0,598	0,717	0,652
GSPO	33486	781	918	746080	0,977	0,973	0,975
GTOUT	332	118	325	780490	0,738	0,505	0,600
GVIO	25222	10424	6278	739341	0,708	0,801	0,751
GVOTE	7047	3128	4139	766951	0,693	0,630	0,660
GWEA	2462	689	1281	776833	0,781	0,658	0,714
GWELF	518	366	1300	779081	0,586	0,285	0,383
MCAT	179921	11998	19017	570329	0,937	0,904	0,921
M11	40312	3044	7090	730819	0,930	0,850	0,888
M12	20668	4477	4636	751484	0,822	0,817	0,819
M13	44068	4471	7970	724756	0,908	0,847	0,876
M131	22217	2911	5025	751112	0,884	0,816	0,848
M132	20726	3185	5327	752027	0,867	0,796	0,830
M14	74927	4704	7972	693662	0,941	0,904	0,922
M141	44501	4259	1699	730806	0,913	0,963	0,937
M142	9934	973	1885	768473	0,911	0,841	0,874
M143	18286	994	3065	758920	0,948	0,856	0,900
Micro-media	2000611	368434	532510	76006210	0,844	0,790	0,816
Macro-media					0,693	0,561	0,579

**Tabla C.9. Valores obtenidos al evaluar *svm1* optimizado para macro-medidas sobre la base de datos LYRL2004 [Lewis et. al, 2004].**

	TP	FP	FN	TN	Prec	Recall	F
CCAT	341605	24316	28936	386408	0,934	0,922	0,928
C11	10482	13494	13169	744120	0,437	0,443	0,440
C12	7658	4639	3905	765063	0,623	0,662	0,642
C13	19386	20447	17077	724355	0,487	0,532	0,508
C14	4644	3878	2606	770137	0,545	0,641	0,589
C15	127925	8249	19681	625410	0,939	0,867	0,902
C151	66896	3088	12628	698653	0,956	0,841	0,895
C1511	16120	3832	6692	754621	0,808	0,707	0,754
C152	55072	14466	16090	695637	0,792	0,774	0,783
C16	1015	693	856	778701	0,594	0,542	0,567
C17	29320	8052	11663	732230	0,785	0,715	0,748
C171	12728	4410	5148	758979	0,743	0,712	0,727
C172	8554	2378	2648	767685	0,782	0,764	0,773
C173	1401	827	1159	777878	0,629	0,547	0,585
C174	5130	449	495	775191	0,920	0,912	0,916
C18	39973	10023	11382	719887	0,800	0,778	0,789
C181	31565	9561	10604	729535	0,768	0,749	0,758
C182	2077	4380	2452	772356	0,322	0,459	0,378
C183	4077	1710	3127	772351	0,705	0,566	0,628
C21	11800	9182	12810	747473	0,562	0,479	0,518
C22	2476	2830	3453	772506	0,467	0,418	0,441
C23	1144	1692	1419	777010	0,403	0,446	0,424
C24	18002	12131	13229	737903	0,597	0,576	0,587
C31	21910	19919	17541	721895	0,524	0,555	0,539
C311	2152	1738	1981	775394	0,553	0,521	0,536
C312	2642	3074	3810	771739	0,462	0,409	0,434
C313	3	4	1071	780187	0,429	0,003	0,006
C32	844	398	1197	778826	0,680	0,414	0,514
C33	8861	8514	6028	757862	0,510	0,595	0,549
C331	805	1351	374	778735	0,373	0,683	0,483
C34	2670	2513	2045	774037	0,515	0,566	0,540
C41	8188	1379	2855	768843	0,856	0,741	0,795
C411	7994	1487	1992	769792	0,843	0,801	0,821
C42	7970	2615	3565	767115	0,753	0,691	0,721
ECAT	88080	16452	28391	648342	0,843	0,756	0,797
E11	4682	3193	3607	769783	0,595	0,565	0,579
E12	16465	12893	9956	741951	0,561	0,623	0,590
E121	1616	817	472	778360	0,664	0,774	0,715
E13	4847	1490	1569	773359	0,765	0,755	0,760
E131	4140	1775	1352	773998	0,700	0,754	0,726
E132	537	308	385	780035	0,636	0,582	0,608
E14	1137	849	975	778304	0,573	0,538	0,555
E141	79	52	285	780849	0,603	0,217	0,319
E142	77	155	115	780918	0,332	0,401	0,363
E143	864	948	308	779145	0,477	0,737	0,579
E21	32795	8002	9080	731388	0,804	0,783	0,793
E211	11245	5882	4116	760022	0,657	0,732	0,692
E212	20793	3399	5759	751314	0,859	0,783	0,820
E31	1273	278	1076	778638	0,821	0,542	0,653
E311	936	284	722	779323	0,767	0,565	0,650
E313	0	0	108	781157	0,000	0,000	0,000



Tabla C.9. Continuación							
CLASE	TP	FP	FN	TN	$\pi$	$\rho$	$F_1$
E41	12526	3662	4060	761017	0,774	0,755	0,764
E411	1389	1077	707	778092	0,563	0,663	0,609
E51	12731	7969	7908	752657	0,615	0,617	0,616
E511	1146	769	1685	777665	0,598	0,405	0,483
E512	8019	4618	4215	764413	0,635	0,655	0,645
E513	1256	62	980	778967	0,953	0,562	0,707
E61	106	92	270	780797	0,535	0,282	0,369
E71	4721	70	381	776093	0,985	0,925	0,954
GCAT	215165	16594	17132	532374	0,928	0,926	0,927
G15	17677	3885	2632	757071	0,820	0,870	0,844
G151	1162	1901	2096	776106	0,379	0,357	0,368
G152	275	635	1797	778558	0,302	0,133	0,184
G153	1604	1437	697	777527	0,527	0,697	0,601
G154	6806	2281	1460	770718	0,749	0,823	0,784
G155	70	149	2016	779030	0,320	0,034	0,061
G156	0	0	258	781007	0,000	0,000	0,000
G157	990	265	1001	779009	0,789	0,497	0,610
G158	2346	2097	1902	774920	0,528	0,552	0,540
G159	0	0	38	781227	0,000	0,000	0,000
GCRIM	24117	6336	6969	743843	0,792	0,776	0,784
GDEF	4885	2193	3724	770463	0,690	0,567	0,623
GDIP	27307	11269	9428	733261	0,708	0,743	0,725
GDIS	6312	2141	2052	770760	0,747	0,755	0,751
GENT	2180	1619	1515	775951	0,574	0,590	0,582
GENV	4099	3387	1990	771789	0,548	0,673	0,604
GFAS	3	5	304	780953	0,375	0,010	0,019
GHEA	3478	1810	2355	773622	0,658	0,596	0,625
GMIL	12978	4133	3792	760362	0,758	0,774	0,766
GOBIT	345	744	486	779690	0,317	0,415	0,359
GODD	839	2612	1873	775941	0,243	0,309	0,272
GPOL	41782	18456	13449	707578	0,694	0,756	0,724
GPRO	669	288	4663	775645	0,699	0,125	0,213
GREL	1787	1872	970	776636	0,488	0,648	0,557
GSCI	1636	949	737	777943	0,633	0,689	0,660
GSPO	33597	844	807	746017	0,975	0,977	0,976
GTOUR	377	277	280	780331	0,576	0,574	0,575
GVIO	22177	5973	9323	743792	0,788	0,704	0,744
GVOTE	5731	1683	5455	768396	0,773	0,512	0,616
GWEA	2232	470	1511	777052	0,826	0,596	0,693
GWELF	486	305	1332	779142	0,614	0,267	0,373
MCAT	180016	12080	18922	570247	0,937	0,905	0,921
M11	42153	4914	5249	728949	0,896	0,889	0,892
M12	20868	4847	4436	751114	0,812	0,825	0,818
M13	43963	4363	8075	724864	0,910	0,845	0,876
M131	21404	1968	5838	752055	0,916	0,786	0,846
M132	20457	2802	5596	752410	0,880	0,785	0,830
M14	75176	4889	7723	693477	0,939	0,907	0,923
M141	43785	3176	2415	731889	0,932	0,948	0,940
M142	10296	1290	1523	768156	0,889	0,871	0,880
M143	18388	1072	2963	758842	0,945	0,861	0,901
Micro-media	2034167	420826	498954	75953818	0,829	0,803	0,816
Macro-media					0,656	0,610	0,619

